

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO E SISTEMAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
DE PRODUÇÃO

RACIOCÍNIO BASEADO EM CASOS PARA AVALIAÇÃO DE PLANOS DE
ROTAS

DISSERTAÇÃO APRESENTADA AO RPOGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA DE PRODUÇÃO DA UNIVERSIDADE
FEDERAL DE SANTA CATARINA PARA OBTENÇÃO
DO GRAU DE MESTRE EM ENGENHARIA

SARAJANE MARQUES PERES

FLORIANÓPOLIS, AGOSTO DE 1999

**RACIOCÍNIO BASEADO EM CASOS PARA AVALIAÇÃO DE PLANOS DE
ROTAS**

SARAJANE MARQUES PERES


**ESTA DISSERTAÇÃO FOI JULGADA ADEQUADA PARA OBTENÇÃO DO
TÍTULO DE MESTRE EM ENGENHARIA.**

**ESPECIALIDADE EM ENGENHARIA DE PRODUÇÃO E APROVADA EM SUA
FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
PRODUÇÃO**

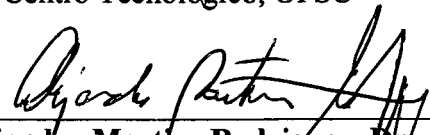


Ricardo Miranda Barcia, Ph. D

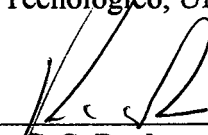
BANCA EXAMINADORA



Lia Caetano Bastos, Dra – Orientadora
Centro Tecnológico, UFSC



Alejandro Martins Rodriguez, Dr
Centro Tecnológico, UFSC



Roberto C. S. Pacheco, Dr
Centro Tecnológico, UFSC

FLORIANÓPOLIS, AGOSTO DE 1999

AGRADECIMENTOS

À Deus, por permitir que tudo isto acontecesse.

A meus familiares pelo apoio incondicional e pela compreensão da necessidade da distância para que este trabalho se realizasse.

À Lia por orientar e apoiar o desenvolvimento deste trabalho.

Aos professores do Programa de Pós Graduação da Engenharia de Produção por proporcionar a aquisição de conhecimento, indispensável para a minha formação de Mestre.

Em especial aos professores Rosina Weber, Roberto Pacheco, Alejandro Martins e Aran Tcholakian por incentivar a continuidade deste trabalho, quando de sua concepção em um trabalho para as disciplinas de Raciocínio Baseado em Casos e Algoritmos Genéticos.

Aos meus amigos que sempre me incentivaram e confiaram na minha capacidade.

Em especial, aos amigos do Departamento de Informática da Universidade Estadual do Oeste do Paraná que, não mediram esforços para suprir minha ausência, necessária para o desenvolvimento deste trabalho.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Problema de Planejamento de Rotas	1
1.2	Motivação	2
1.3	Objetivos do trabalho.....	5
1.4	Estrutura do trabalho	5
2	RACIOCÍNIO BASEADO EM CASOS E SISTEMAS HÍBRIDOS.....	6
2.1	Considerações Gerais	7
2.2	Raciocínio Baseado em Casos.....	8
2.2.1	Conceitos	8
2.2.2	Classes	9
2.2.3	Fases	11
2.2.3.1	Identificação e representação de um caso	11
2.2.3.2	Busca e recuperação de casos similares	13
2.2.3.3	Adaptação / Justificativa	15
2.2.3.4	Avaliação e reparo.....	17
2.2.3.5	Aprendizado e esquecimento.....	17
2.2.4	Vantagens e desvantagens	18
2.2.5	Aplicabilidade	19
2.3	Sistemas Híbridos.....	19
2.3.1	Considerações gerais	19
2.3.2	Classificações	20
2.4	Aplicações e Exemplos	23
3	PLANEJAMENTO DE ROTAS	27
3.1	Considerações gerais	27
3.2	Divisão e Definições	28

3.3	Abordagens Tradicionais.....	31
3.4	Fatores Importantes no Planejamento de Rotas.....	34
3.5	Exemplos de aplicação	36
4	PROTEUS	40
4.1	Introdução.....	40
4.2	Arquitetura híbrida.....	40
4.3	A operação do sistema.....	43
4.4	O Módulo Genético.....	44
4.5	O módulo de RBC.....	46
4.5.1	Representação do Conhecimento	46
4.5.1.1	A representação da malha.....	47
4.5.1.2	Os dados qualitativos – características dos perfis de usuário X dos segmentos de rua	48
4.5.1.3	As bibliotecas de casos.....	49
4.5.1.4	O conjunto de procedimentos.....	51
4.5.1.5	Indexação.....	52
4.5.2	Recuperação	52
4.5.2.1	Métrica de similaridade.....	52
4.5.2.2	Recuperação	55
4.5.2.3	Best Match.....	55
4.5.3	Adaptação.....	55
4.5.4	Avaliação e aprendizado	56
5	SIMULAÇÕES	58
5.1	Considerações gerais	58
5.2	O módulo genético	59
5.3	O módulo de raciocínio baseado em casos	62
6	CONCLUSÃO	70

6.1 Considerações Finais.....	70
6.2 Trabalhos futuros.....	72
REFERÊNCIAS BIBLIOGRÁFICAS	74
ANEXO 1 - DIAGRAMA ENTIDADE RELACIONAMENTO.....	80
ANEXO 2 - ALGORITMO	82
ANEXO 3 – ARTIGO PUBLICADO REFERENTE AO TRABALHO.....	86

LISTA DE FIGURAS

Figura 2.1: Fases do processo de RBC.....	11
Figura 2.2: Representação de um caso, Hammond, 1989 apud (KOLODNER, 1993, p. 172).....	13
Figura 2.3: Modelos de integração de sistemas inteligentes [Medsker and Bailey, 1992] apud (Medsker, 1995, p.21).	21
Figura 2.4: Visão geral da integração de tecnologias convencionais e inteligentes (MEDSKER, 1995, p. 229).	23
Figura 4.1: O processo de RBC integrado ao uso ao uso de Algoritmo Genético, adaptada de (MEDSKER, 1995, p. 216).	41
Figura 4.2: Arquitetura do Sistema (PERES, 1997)	42
Figura 4.3: Tela inicial do sistema PROTEUS.	43
Figura 4.4: Rota representada na forma de um cromossomo.....	45
Tabela 4.1: Descrição dos atributos dos segmentos.	47
Figura 4.5: Exemplo de representação da malha viária.	48
Figura 4.6: Perfis de usuários.	49
Figura 4.7: Associações entre Perfis de Usuários e Características Descritivas dos Segmentos.	49
Figura 4.8: Formato dos casos da biblioteca.	50
Figura 4.9: Exemplo para análise de rotas	54
Figura 5.1: Centro de Cascavel [www.zaz.com.br/verão].	58
Figura 5.2: Pontos de origem e destino da rota de teste do AG.	59
Figura 5.3: Primeira rota gerada.....	60
Figura 5.4: Três rotas geradas	60
Figura 5.5: Cinco rotas geradas.....	61
Figura 5.6: Todas as rotas geradas.	62
Figura 5.7: Rotas com o trecho final diferente.....	62
Figura 5.8: Tentativa de recuperação completa.	63
Figura 5.9: Tentativa de recuperar a rota como um trecho de outra.	64
Figura 5.10: Similaridade das três rotas escolhidas.	64
Figura 5.11: Rotas escolhidas.....	65
Figura 5.12: Primeiro caso recuperado.	65

Figura 5.13: Valor de Similaridade do primeiro caso recuperado.	66
Figura 5.14: Primeiro caso recuperado que envolve a rota procurada.	67
Figura 5.15: Rota recuperada parcialmente.....	67
Figura 5.15: Exemplo de rota e sua similaridade.	68
Figura 5.16: Características dos segmentos que compõe a rota.	68
Figura 5.17: Aprendizado do sistema.	69

RESUMO

A tarefa de planejar rotas em um ambiente urbano usualmente requer um trabalho de raciocínio flexível, adaptativo e capaz de lidar com a ausência ou “incompletude” de informações. Com o aumento da complexidade do ambiente urbano, o planejamento de rotas torna-se penoso e estressante, e a dinamicidade com a qual as pessoas se movem entre cidades, faz com que os planejadores lidem com ambientes desconhecidos, gerando planos, muitas vezes, ineficientes e/ou ineficazes.

Neste sentido, faz-se desejável, a automatização do processo de planejamento de rotas, de forma que estas sejam formuladas com o máximo de eficiência e eficácia no que diz respeito à sua executabilidade e às preferências de cada pessoa.

Assim, as técnicas de Inteligência Artificial surgem como ferramentas que permitem suprir algumas deficiências encontradas nas técnicas convencionais, no que diz respeito a: tratamento de informações vagas e imprecisas e aproximação do processo de raciocínio automático ao processo de raciocínio humano.

Neste contexto, gerou-se um sistema híbrido para criar planos de rotas e avaliá-los de acordo com sua adaptabilidade a diferentes restrições, o qual foi implementado utilizando duas técnicas de Inteligência Artificial: Algoritmos Genéticos e Raciocínio Baseado em Casos (RBC).

Esse trabalho apresenta o desenvolvimento deste sistema (Sistema PROTEUS - Planejador de ROTas para Transportes em Espaços Urbanos e Similares), enfocando a utilização de RBC no processo de obtenção das rotas.

ABSTRACT

The task of planning routes in a urban environment requires a flexible and adaptative reasoning, which works with incompleteness and vagueness of information. For this purpose it has to be consider the complexity of urban environment, therefore the planning routes becomes laborious and stress, and the dynamic movement of the people among cities makes with the planners handles with unknown environment, getting some times inefficient and/or inefficacious plans.

By this way, is interesting to get an automation process of planning routes which considers the variables of efficiency and efficacy, as well the preferences of each peoples.

Thus, Artificial Intelligence rise as a tool that allows to resolve deficiencies in a tradicional way, about: treatment of vagues and unaccuracy informations; closed process of automatic reasoning to process of human reasoning.

In this context, a hybrid system was developed mainly to get routes planning and evaluate them considering differents constraints. So, the system was implemented using two techincs of Artificial Intelligence: Genetic Algorithms (GA) and Case Based Reasoning (CBR).

This work presents the development of Hybrid System (PROTEUS)¹ which foccus the use of CBR in a process to get routes.

¹ PROTEUS – Planejador de ROTas para Transporte em Espaços Urbanos e Similares.

1 INTRODUÇÃO

1.1 PROBLEMA DE PLANEJAMENTO DE ROTAS

O problema de Planejamento de Rotas pode ser encontrado na literatura com diferentes denominações, variando de acordo com o enfoque que lhe é dado. Em GOLDEN (1988), esse problema é denominado “roteamento de veículos”, já que seu objetivo é, basicamente, encontrar rotas para um frota de veículos de entrega. Em HAIGH (1995), onde o objetivo é traçar uma rota, o problema é denominado “planejamento de rotas”. Este problema, está também relacionado, na literatura, aos clássicos problemas do “caixeiro viajante” (LAWLER, 1985), do “carteiro-chinês” (LEE, 1994), entre outros.

Para o enfoque dado neste trabalho, o problema de planejamento de rotas pode ser definido como um problema de encontrar um caminho válido dentro de uma malha viária urbana, que seja satisfatório para um usuário com características e preferências específicas. Trata-se portanto, em sua natureza, de um problema matemático cuja formulação é simples porém, sua resolução é bastante problemática, assim como os clássicos problemas que compõem a classe onde se encontram problemas de otimização tais como o caixeiro viajante, o carteiro chinês e outros.

Os estudos sobre o problema de Planejamento de Rotas são realizados, já a algumas décadas, por pesquisadores da área de Transportes, Matemática e Ciência da Computação. Dentro destes estudos, várias soluções para diferentes variações desta classe de problemas foram propostas e, a cada solução nova, grandes melhorias de performance foram atingidas e cada vez mais, a complexidade dos problemas enfocados vêm aumentando.

Os pesquisadores da área de Transportes são importantes na especificação do problema e de suas restrições, atuando como validadores¹ dos resultados apresentados pelos sistemas construídos para auxiliar em problemas desta natureza.

Os pesquisadores das áreas de Matemática e Ciência da Computação unem esforços para encontrar formas de resolver os problemas desta categoria, de maneira eficiente², concentrando seus esforços em dois objetivos principais: diminuir o tempo de espera pela resposta do sistema e aumentar a capacidade do sistema no sentido de resolver problemas cada vez mais complexos e abrangentes.

Neste trabalho, o enfoque é aplicar novas técnicas para resolver este problema, analisando a viabilidade da aplicação destas para alcançar um resultado satisfatório aos usuários do sistema, contribuindo para o alcance dos objetivos acima citados.

1.2 MOTIVAÇÃO

A agilidade, praticidade, objetividade e confiabilidade exigida, atualmente, na vida dentro de grandes cidades, leva a uma questão extremamente importante e sob certas circunstâncias, muito estressante, envolvendo problemas de tráfego. Malhas viárias complexas e grande volume de veículos e pessoas nestas malhas, tornam a simples tarefa de ir e vir de forma satisfatória, algo complicado e aflitivo, como mostra a pesquisa realizada em STATHOPOULOS et al (1994), onde alguns motoristas pesquisados planejaram rotas inválidas por desconhecer características como ruas de mão única e retornos proibidos.

Essa complexidade justifica o desenvolvimento de sistemas automatizados que auxiliem as pessoas, de forma rápida, precisa e adequada, na formulação de soluções para minimizar os esforços despendidos dentro do trânsito. Existem vários programas

¹ Segundo PRESSMAN (1995), validação é o conjunto de atividades que garantem que o software que foi construído é fiel às exigências do cliente. No problema em questão, os clientes são, em sua maioria, pesquisadores da área de Transportes que, solicitam aos pesquisadores da área de Computação e Matemática, a automação otimizada de alguma atividade relacionada a transportes, cooperando com informações especialistas sobre o problema.

² Também segundo PRESSMAN (1995), a eficiência de um software está relacionada a quantidade de recursos computacionais e de código exigida para que um programa execute sua função, ou seja, uma execução eficiente é aquela que utiliza os recursos computacionais tão bem quanto possível.

destinados a esta tarefa, apresentando diferentes enfoques. Alguns são comerciais, facilitando a interface com o usuário e necessitando de equipamentos de alta tecnologia para sua operação, como o Route Planner (INFOTEL), outros são mais acadêmicos, e têm como objetivo principal, abrir novos caminhos dentro de técnicas inovadoras ou melhorar as soluções já existentes, como o PULSAR (BERSTEIN, 1998), (SCOTT, 1997a), (SCOTT, 1997b).

Tradicionalmente, a resolução do traçado de rotas tem sido tratada pelas áreas de Transportes, Pesquisa Operacional e Teoria dos Grafos. Mais recentemente, esse problema vem sendo resolvido utilizando-se técnicas de Inteligência Artificial, como pode ser observado nos seguintes trabalhos:

- redes neurais (DURBIN, 1997), (BURKE, 1992);
- algoritmos genéticos (MÜHLENBEIN, 1991);
- raciocínio baseados em casos (HAIGH, 1995);
- sistemas híbridos (NAKAMITI, 1994).

Considerando o problema de criar planos de rotas, parece perfeitamente natural a afirmação:

... se alguém quer ir a algum lugar onde já esteve antes, e esse alguém não quer usar um mapa para encontrar uma nova rota, ele usará a rota anterior que está em sua memória para guiar a viagem atual. Se ele quer ir próximo de onde ele já esteve antes, ele estenderá sua velha solução para adaptá-la ao novo problema (LIU, 1996, p. 02) trad. pela autora.

Assim sendo, intuitivamente, o uso de Raciocínio Baseado em Casos (RBC) parece se encaixar perfeitamente para a resolução deste problema, auxiliando na tomada de decisão, já que a essência desta técnica é a utilização de experiências passada para resolver problemas atuais.

Além disso, algumas pesquisas levantam uma questão interessante sobre a modelagem de tomada de decisão. Em GÄRLING (1994), o autor comenta que em modelagens de tomada de decisão, existe clara distinção entre teorias estatísticas (sobre as quais são estimados parâmetros para realização de escolhas e julgamentos), e a teorias substanciais, presentes em ciências comportamentais, e que deve guiar qualquer modelagem de um processo do mundo real. Considerando que RBC é uma técnica que tenta imitar o raciocínio humano, é interessante tentar se aproximar do processo de tomada de decisão humano utilizando-a e, minimizando o tecnicismo das modelagens

matemáticas.

Porém, existem alguns pontos a serem observados, considerando o problema proposto, que são discutidos mais detalhadamente nos capítulos seguintes:

- um sistema de RBC, trabalha com experiências armazenadas em uma base de casos que, inicialmente, está vazia, a menos que de alguma forma, sejam inseridos casos iniciais (a partir de experiências de um especialista, por exemplo), porém isto seria um incômodo, pois a cada mudança de ambiente (mudança de cidade) este trabalho deveria ser refeito;
- a diversidade de experiências possíveis neste ambiente, exigiria uma quantidade muito grande de casos para abranger todo esse escopo de atuação, sendo possível que algumas situações ficassem excluídas;
- mudança na malha viária invalidariam algumas experiências armazenadas, e uma nova parte de situações possíveis estaria novamente fora do alcance da técnica;

Observa-se então, que existe um problema de geração de rotas iniciais ou rotas totalmente diferentes das existentes. Uma técnica que fosse capaz de criá-las e de preferência, criar várias soluções para uma única situação, poderia fazer o trabalho de alimentação da biblioteca de casos do RBC, fomentando-a cada vez que uma situação totalmente nova, diferente de qualquer outra que já tenha ocorrido, aparecer.

A técnica de Algoritmos Genéticos parece atender muito bem ao problema de geração de rotas, já que é uma técnica que pode criar várias soluções para um determinado problema.

Portanto, poder-se-ia pensar em resolver o problema apenas com esta técnica, porém, existem algumas características interessantes a serem analisadas que, a princípio, dificultaria a representação do problema dentro da técnica de Algoritmos Genético e aumentaria o esforço computacional exigido pela mesma.

Cada uma das duas técnicas apresentadas acima, pode resolver o problema proposto dentro de suas limitações, apresentando alguma dificuldade em satisfazer todas as restrições existentes em situações reais. Neste sentido, uma direção a ser seguida é utilizar a tecnologia de Sistemas Híbridos, aproveitando o máximo de cada técnica e suprimindo as falhas de uma, através do uso da outra.

1.3 OBJETIVOS DO TRABALHO

O objetivo geral desse trabalho é desenvolver um sistema eficiente para gerar rotas válidas que atendam as características e preferências do usuário.

Como objetivos específicos tem-se:

- Implementar uma entidade fornecedora de rotas para alimentação da base de casos utilizada pela técnica de Raciocínio Baseado em Casos. A descrição desta implementação foi desenvolvida como um outro trabalho correlacionado (CASTRO, 1999);
- Implementar uma entidade avaliadora de uma solução adequada as características e preferências do usuário;
- fornecer rotas alternativas para uma mesma situação;
- adaptar as rotas à diferentes perfis de usuários;
- criar um sistema independente da malha viária.

1.4 ESTRUTURA DO TRABALHO

Este trabalho está dividido em seis capítulos.

No primeiro capítulo é feita a identificação do problema, a descrição da motivação para seu desenvolvimento, o delineamento dos objetivos pretendidos e sua estrutura.

RBC e Sistema Híbridos são os assuntos abordados no capítulo dois, onde uma breve explanação sobre os conceitos e definições inerentes a cada uma destas técnicas é realizada. Alguns exemplos de trabalhos sobre Sistemas Híbridos envolvendo Raciocínio Baseado em Casos são apresentados.

A seguir, no terceiro capítulo, são apresentados e discutidos os modelos tradicionais utilizados para o problema de planejamento de rotas.

No quarto capítulo é descrito o sistema PROTEUS. Neste capítulo uma explicação detalhada do módulo RBC é apresentada enquanto que o módulo genético é apresentado sucintamente³.

³ O Módulo Genético é parte do Trabalho de Pesquisa de Josué Pereira de Castro realizado no Programa de Pós-Graduação em Engenharia de Produção e Sistemas da Universidade Federal de Santa Catarina.

No quinto capítulo, simulações utilizando o sistema desenvolvido são apresentadas.

No último capítulo, as conclusões e recomendações para futuros trabalhos são apresentadas.

Finalmente, é listada a bibliografia utilizada, bem com, citada neste trabalho.

Nos anexos 1, 2 e 3, estão disponíveis, respectivamente, um detalhamento sobre a base de dados do sistema, o algoritmo utilizado e a cópia do artigo publicado referente a este trabalho.

2 RACIOCÍNIO BASEADO EM CASOS E SISTEMAS HÍBRIDOS

2.1 CONSIDERAÇÕES GERAIS

Se um organismo carrega um **modelo em pequena escala** da realidade externa e as suas possíveis ações dentro de sua cabeça, ele é capaz de experimentar várias alternativas, concluir qual delas é a melhor, reagir às situações futuras antes que elas cheguem, utilizar o conhecimento de eventos passados para lidar com o presente e o futuro e sempre agir de forma mais competente e segura em emergências com as quais se depara Craik, 1943 apud (RUSSEL, 1995, p. 13).

Na década de 70, Roger Schank desenvolveu alguns trabalhos que utilizavam o raciocínio baseado em memória SHANK e ADELSON apud (CARVALHO, 1998). Anos depois, na década de 80, o mesmo pesquisador publicou um trabalho envolvendo “memória dinâmica”, SCHANK apud (CARVALHO, 1998). Estes trabalhos podem ser considerados os pioneiros no desenvolvimento da técnica de Raciocínio Baseado em Casos (RBC). Mais tarde, essa técnica foi expandida pelos estudos dos alunos de Roger Schank, com um destaque especial a Janet Kolodner, que desenvolveu o primeiro sistema utilizando RBC (CARVALHO, 1998). Esse sistema serviu de base para muitos outros, tais como: MEDIATOR (área de mediação), PERSUADER (área de persuasão), CHEF e JULIA (área de culinária), CASEY (área de medicina), PROTOS, GREBE e HYPO (área de direito), (KOLODNER 1993).

A tecnologia de Sistemas Híbridos (SH) que integra duas ou mais técnicas na resolução de problemas, vem sendo amplamente estudada, pelo menos, segundo MEDSKER (1995), desde o final da década de oitenta e tem crescido rapidamente. Os primeiros trabalhos desenvolvidos envolviam as técnicas de Redes Neurais e Sistemas Especialistas, porém, atualmente, uma variedade de combinações, vem surgindo e se mostrando extremamente eficientes.

Este capítulo é dividido em seções, sendo que na seção 2.2 apresenta-se definições e conceitos básicos sobre RBC acompanhados de alguns exemplos. As

questões relacionadas a SH estão na seção 2.3 e na última seção são apresentados algumas aplicações de RBC em SH.

2.2 RACIOCÍNIO BASEADO EM CASOS

2.2.1 CONCEITOS

Várias definições para a técnica de Raciocínio Baseado em Casos (RBC) são encontradas na literatura. Entre elas estão:

- RBC adapta velhas soluções para encontrar novos usos, utilizando casos passados para explicar ou criticar novas situações, raciocina a partir de precedentes para interpretar uma nova situação ou cria uma solução equivalente para um novo problema (KOLODNER, 1993, p. 04).
- RBC tenta encontrar a solução para um problema atual, procurando por um problema similar em uma base de experiências, tomando a solução do passado como ponto inicial para encontrar uma solução para o problema atual (RICHTER, 1998, p. 05).

Basicamente, RBC é uma metodologia da Inteligência Artificial que emula o processo de raciocínio humano onde, experiências passadas são utilizadas na resolução de problemas atuais.

Esta técnica está sendo amplamente utilizada devido a simplicidade e plausibilidade do processo inerente a ela. O processo pode ser resumidamente dividido na seguintes etapas:

- identificação e análise da situação atual (um problema a ser resolvido);
- busca e recuperação de uma situação passada similar à situação atual;
- análise, modificação - se necessário - e aplicação do conhecimento contido na situação recuperada para resolver o problema atual;
- armazenamento da situação atual e de sua solução, se desejável.

Este processo é simples porque permite que um problema seja resolvido por um indivíduo, sem que esse tenha conhecimento específico sobre o domínio onde está trabalhando e com um mínimo esforço, apenas repetindo ou adaptando o que já foi realizado no passado.

É plausível por emular o comportamento humano diante de situações novas,

respaldando-se em deduções empíricas, as quais acreditam que problemas similares são resolvidos de maneiras similares e que esses problemas têm grande probabilidade de ocorrerem novamente de forma parecida.

As experiências passadas são as entidades denominadas “casos”. Um caso é a representação de uma situação/experiência através de atributos-valores, que descrevem o conteúdo e/ou o contexto dessas. Um conjunto de casos representa o conhecimento que se possui sobre um determinado domínio, e é sobre esse conjunto que todo o processo de RBC se desenvolve.

Tanto a escolha dos atributos que melhor descreverão os casos quanto a forma como eles estarão organizados, são complexas e variam conforme o domínio da aplicação onde a técnica está sendo utilizada e conforme o objetivo para o qual ela se destina.

É importante salientar que o caso descreve uma situação de acordo como ela ocorreu. Em RICHTER (1998), os autores salientam que um caso, em sua forma primária, reflete uma situação única, não havendo procedimento de generalização ou incorporação de lições aprendidas, explicações de como a solução foi obtida ou em quais princípios está baseada. Porém, os efeitos causados pelas soluções representadas podem estar descritos. Os conhecimentos abstraídos, deduzidos ou inferidos devem estar armazenados na medida de similaridade ou nas regras de adaptação da solução, discutidos posteriormente neste capítulo.

Outro grande atrativo deste processo é que ele fornece uma continuidade do conhecimento através do armazenamento, uso e aprimoramento de experiências passadas.

2.2.2 CLASSES

A técnica de RBC encontra-se dividida em dois grandes propósitos de aplicação (KOLODNER, 1993):

- RBC para resolução de problemas;
- RBC interpretativo.

No RBC para solução de problemas, situações prévias são utilizadas para propor uma solução para um novo problema. As similaridades e diferenças existentes entre os casos anteriores e o novo caso, podem ser utilizadas para determinar quando uma

adaptação é necessária, guiando o processo adaptativo da solução. Esta classe de RBC suporta, principalmente, tarefas de projeto, planejamento e explicação/diagnóstico.

Tarefas de projeto são aquelas tarefas que são definidas como um conjunto de restrições onde o resolvidor de problemas é requerido para fornecer um artefato concreto que satisfaça as restrições, KOLODNER (1993).

As tarefas de projeto podem ser divididas em duas categorias, RICHTER (1998):

- tarefas de configuração: entendida como a construção de um artefato a partir de um conjunto de componentes, tal que certas condições sejam satisfeitas;
- tarefas de projeto, onde um certo grau de criatividade deve ser inserido porque alguns componentes do artefato podem não estar disponíveis *a priori* e dependendo do grau de criatividade exigido, uma subdivisão pode ser estabelecida: projeto de rotina, projeto inovativo e projeto criativo.

Em KOLODNER (1993), planejamento é definido como processo de estabelecer uma seqüência de passos ou um escalonamento para alcançar algum estado de mundo.

Segundo Shank, 1986 apud (KOLODNER, 1993), uma explicação pode ser realizada, através de RBC, apoiada na lembrança de fenômenos similares, emprestando suas explicações e adaptando-as para a situação atual. Diagnóstico é um tipo particular de problema de explicação, onde o problema a ser resolvido é dado como um conjunto de sintomas (coisas que não são como deveriam ser) que devem ser explicados. Um RBC para diagnóstico usa casos para sugerir explicações para os sintomas ou prevenir de explicações inapropriadas encontradas no passado.

No RBC interpretativo, o objetivo é justificar ou classificar uma nova situação através de comparação com casos que já foram julgados ou classificados. Esta classe suporta, principalmente, tarefas de justificativa e raciocínio adverso e tarefas de classificação e interpretação.

Como tarefas de raciocínio adverso entende-se a argumentação persuasiva para convencer outros que, nós ou nossas posições estão corretas, através da elaboração de argumentos e justificativas. Já tarefas de interpretação, no contexto de RBC, significa decidir se um conceito pertence ou não a uma classe, podendo exigir a redefinição de uma velha situação em termos de novas dimensões, para criar um novo padrão de classe quando um adequado não existe.

2.2.3 FASES

O processo de RBC pode ser melhor entendido a partir de um detalhamento de cada uma de suas fases. A Figura 2.1 mostra a sequência das fases do processo.

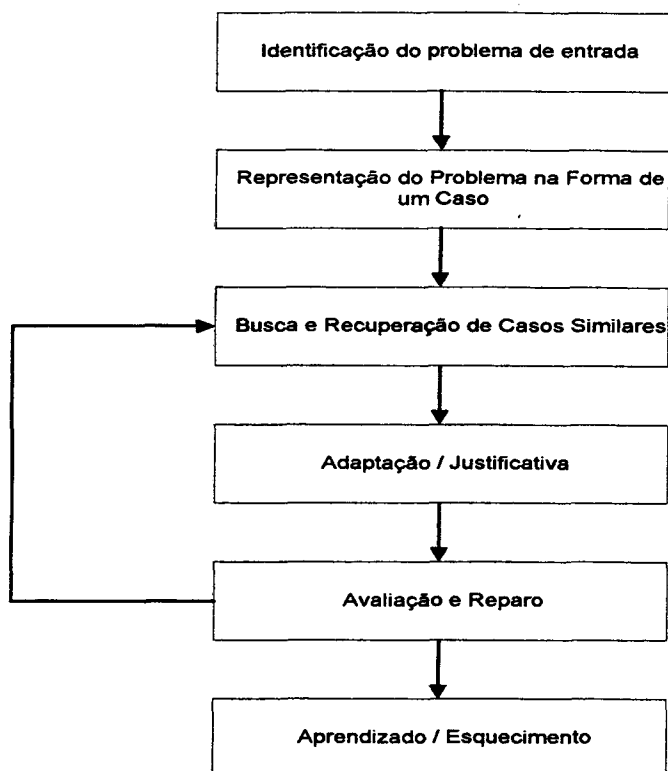


Figura 2.1: Fases do processo de RBC

2.2.3.1 IDENTIFICAÇÃO E REPRESENTAÇÃO DE UM CASO

A primeira fase do processo trata da identificação do problema -caso de entrada-, onde delimita-se quais são as características que descrevem a situação, partindo em seguida para a representação desta, na forma de um caso.

O caso é a estrutura que representa uma experiência passada, descrevendo-a quanto ao que foi verdadeiro ou falso, correto ou incorreto, bom ou ruim, útil ou não. O conjunto de vários casos, que forma uma base ou biblioteca de casos, é o centro de um sistema de RBC.

Segundo KOLODNER (1993), um caso é um pedaço de conhecimento contextualizado que representa uma experiência. Ele ensina uma lição fundamental para alcançar as metas de um raciocinador, sendo dividido em duas partes principais: a lição que ele ensina e o contexto no qual ele pode ensinar sua lição.

Em RICHTER (1998), define-se um caso como sendo uma entidade que permite recordar um episódio onde um problema ou uma situação problema foi totalmente ou parcialmente resolvida.

O formato dos casos e a maneira como serão armazenados devem ser definidos no momento da construção do sistema. A representação dos casos envolve basicamente três aspectos:

- como descrever os casos: escolha dos atributos descritivos;
- como organizá-los na memória: utilizando redes semânticas (KOLODNER, 1993, p. 417), *scripts* (KOLODNER, 1993, p. 10), “MOPS” (*memory organization packets*) (KOLODNER, 1993, p. 108), entre outras formas de representação;
- como indexá-los: escolha de índices, entre os atributos descritivos, que melhor identificam os casos, proporcionando uma recuperação eficiente.

A descrição de um caso é, usualmente, dividida em duas partes (WBEBER, 1997):

- descrição do problema: onde as características descritivas podem ter a forma de nomes, números, funções ou textos, descrevendo características, objetivos, metas, restrições etc;
- descrição da solução: onde é apontada a solução do problema de entrada, sendo que sua forma depende do propósito do RBC, podendo ser um nome, texto, número etc.

Dentro da problemática de descrição de um caso, levanta-se a questão sobre o que descrever. Basicamente deve-se descrever o problema e a sua solução porém, em KOLODNER (1993) é incluído um terceiro aspecto: o resultado da aplicação da solução ao problema. A escolha por descrever os três aspectos depende do domínio da aplicação e de seu propósito.

Em KOLODNER (1993), encontram-se vários exemplos de como casos podem ser estruturados e de como situações diversas podem ser representadas na forma de um caso. A Figura 2.2 apresenta um exemplo de um caso que representa uma situação que pode ser utilizada no planejamento de um cardápio no sistema CHEF's, Hammond, 1989 apud (KOLODNER, 1993, p. 172):

Outra questão importante na representação de casos é a escolha de índices através dos quais, buscas serão realizadas.

Segundo KOLODNER (1993), o problema de indexação de casos tem várias partes:

- trata-se da designação de *labels* para os casos que são colocados na base de casos, afim de assegurar que eles sejam recuperados nas horas apropriadas. Em geral, esses *labels* determinam sob quais circunstâncias o caso pode ser utilizado na resolução de um novo problema;
- trata-se do problema de organizar os casos tal que a busca realizada sobre a base de casos possa ser feita de forma eficiente;

Portanto, os índices escolhidos representarão o contexto de cada caso e sua utilidade dentro do domínio de conhecimento no qual está inserido, sendo essa escolha, uma tarefa essencial para o processo de RBC.

Problema:	(incluir tofu) (prato quente) (estilo mexido)
Solução:	(ingredientes ingr1 (tofu kg .5) ingr2 (molho de soja colher de sopa 2) ingr3 (vinagre colher 6) ingr4 (fécula de milho colher de sopa .5) ingr5 (açúcar colher 1) ingr6 (brócolis kg 1) ingr7 (pimenta pedaço 6) (ações ação1 (cortar objeto (ingr1) tamanho (pedaço grande)) ação2 (marinar objeto (resultado ação1) com (& (ingr2) (ingr3) (ingr4) (ingr5)) tempo (20)) ação3 (cortar objeto (ingr6) tamanho (pedaço grande)) ação4 (fritar objeto (& (resultado ação2) (ingr7)) tempo (1)) ação5 (adicionar objeto (resultado ação3) para (resultado ação4)) ação6 (fritar objeto (resultado ação5) tempo (2))) (estilo mexido)

Figura 2.2: Representação de um caso, Hammond, 1989 apud (KOLODNER, 1993, p. 172)

2.2.3.2 BUSCA E RECUPERAÇÃO DE CASOS SIMILARES

Quando já se possui uma base/biblioteca de casos, o processo de busca e recuperação de casos a serem utilizados para resolver situações atuais pode ser iniciado.

Nesta etapa, a busca por casos similares é realizada através de algoritmos que

varrem a biblioteca de casos analisando-os de acordo com o quanto são similares ao caso de entrada. O que faz com que um caso seja similar a outro, difere de acordo com o domínio e o propósito da aplicação, porém, num contexto geral, pode-se dizer que a semelhança entre casos está na similaridade das características que representam o conteúdo e o contexto das experiências em questão (passadas e atuais).

A questão de similaridade é bastante singular e precisa ser estudada para cada domínio. A similaridade entre casos pode ser determinada por métricas (resultado de uma função), por metas, restrições ou classificações, sendo que nos três últimos casos, uma métrica pode ser utilizada para estabelecer o grau de similaridade dos casos (WEBER, 1997). Alguns autores dividem a avaliação da similaridade em dois grupos:

- similaridade sintática: comparação sintática dos valores dos atributos, analisando sinônimos, categorias ordinais, intervalos, etc;
- similaridade semântica: abrange o significado dos casos relacionando-os aos valores dos atributos, envolvendo o processamento de linguagem natural.

Tanto a similaridade sintática quanto a semântica, quando aplicadas em problemas, retornam valores que medem a similaridade entre os casos. Esta medida de similaridade é encontrada através de funções e/ou regras que guardam algum conhecimento sobre o domínio do problema, já que para julgar se determinadas situações são parecidas, é necessário saber o que exatamente as tornam parecidas. As funções numéricas utilizadas podem ser, entre outras, “o vizinho mais próximo” ou uma média ponderada.

Depois de encontrados os casos similares, é necessário estabelecer uma restrição para recuperação. Um limiar de similaridade (recuperação apenas de casos cuja medida de similaridade ultrapassa esse limiar) ou um número limite de casos pode ser estabelecido. Na utilização de métricas de similaridade, todos os casos da biblioteca de casos podem ser avaliados com relação ao caso de entrada, a recuperação pode ser feita utilizando métodos eliminatórios, onde restrições são utilizadas para reduzir o espaço de busca dentro da biblioteca de casos ou ainda por classificação, utilizada em sistemas cujas memórias estão estruturadas em classes ou categorias.

Para finalizar a fase de busca e recuperação, a escolha do melhor caso (*best match*) para ser recuperado deve ser feita de acordo com algum critério pré-estabelecido, utilizando regras, heurísticas ou mesmo interação com o usuário, e este

será o caso escolhido para ser usado na resolução do problema de entrada.

Assim que o *best match* é escolhido e recuperado, ele deve ser analisado de acordo como o propósito da aplicação de RBC. No caso de RBC para resolução de problemas, uma revisão é feita para verificar se é necessário fazer algum tipo de adaptação para então, utilizá-lo para resolver o problema de entrada e, no caso de RBC interpretativo, uma revisão deve ser feita para buscar os argumentos que justifiquem um resultado ou interpretação desejados.

A fase de escolha de um *best match* pode ser suprimida de um processo de RBC, se assim for desejável. Neste caso, um *ranking* de casos deve ser realizado, ordenando os casos de acordo com as medidas de similaridades encontradas ou através da utilização de regras e/ou heurísticas.

Pode ainda ser desejável, a combinação de diversos casos similares para formulação de uma resposta/solução do problema de entrada, neste caso, algumas regras de adaptação deverão ser usadas como descrito na próxima seção.

2.2.3.3 ADAPTAÇÃO / JUSTIFICATIVA

Certas situações, em RBC para resolução de problemas, os casos recuperados podem apresentar uma solução aproximada para o problema atual, exigindo algumas modificações para melhor ajustá-la na resolução deste problema. Estas modificações são chamadas de adaptação e podem ser feitas através da utilização de técnicas especificadas no sistema de RBC, através de regras que representam um conhecimento adicional sobre o domínio do problema ou até mesmo através de interações com o usuário. Esta fase do processo é considerada o “gargalo” da técnica de RBC.

Em KOLODNER (1993), são apresentados dez métodos para realizar adaptação, divididos em três classes:

Métodos por substituição: onde valores apropriados para a nova situação substituem valores encontrados na solução recuperada:

- Reinstanciação: uso de novos valores para determinados atributos;
- Ajuste de parâmetros: uso de heurísticas para ajustar parâmetros numéricos;
- Busca local: uso de uma busca para encontrar uma estrutura de conhecimento auxiliar para substituir uma estrutura inapropriada para a nova situação;
- Memória de consulta: significa procurar algo com uma determinada

descrição na base de casos;

- Busca especializada: procurar novas estruturas de conhecimento através de uma busca guiada por heurísticas;

- Substituição baseada em casos: uso de outros casos para gerar substituições;

Métodos transformacionais: transformam uma solução velha em uma solução que resolverá a nova situação:

- Transformação com senso comum: heurísticas de senso comum são utilizadas para realocar, excluir ou inserir componentes em uma solução;

- Reparo guiado por modelo: transformações guiadas por modelo causal;

Outros métodos:

- Reparo e adaptação de propósito especial: heurísticas são utilizadas para fornecer adaptações específicas do domínio não cobertas por outros métodos;

- Repetição derivacional: uso dos passos envolvidos em soluções antigas, para resolver a situação atual.

Para descobrir a necessidade de uma adaptação e onde exatamente, dentro do caso escolhido, ela se faz necessária, alguns aspectos deve ser analisados:

- diferenças em pontos na especificação dos problemas (recuperado e atual);
- verificação de inconsistências entre a solução proposta e as metas e restrições do novo problema;
- análise das possíveis falhas encontradas na aplicação da solução proposta no passado;
- análise das adaptações que se fizeram necessárias no passado sobre a solução recuperada.

Em RBC interpretativo, se faz necessário a busca de justificativas para o problema atual, a partir dos casos recuperados. A justificativa é, segundo KOLODNER (1993), o processo de criar um argumento para uma solução proposta, através de um processo de comparação da nova situação com o caso recuperado, procurando por similaridades entre as situações e argumentos que justifiquem o resultado desejado ou diferenças que impliquem em fatores que devem ser levados em consideração.

O processo de justificativa pode servir para dois propósitos (KOLODNER, 1993, p. 156):

- um guia para avaliação de uma solução: se parte de uma solução prévia é justificável, uma forma de avaliar se o resultado de um velha solução servirá para uma nova situação é examinar se as propriedades que justificaram partes da velha solução estão presentes no nova caso;
- um guia para a seleção de índices: a justificativa para alguma escolha ou solução diz o que foi importante no mundo do domínio para chegar àquela conclusão. O propósito dos índices é especificar o que é importante em um caso, e o que foi importante no mundo do domínio para resolver um dado problema. Através do processo de justificativa, índices podem ser encontrados.

2.2.3.4 AVALIAÇÃO E REPARO

Após a realização das atividades inerentes às fases já descritas, a solução deve ser aplicada ao problema de entrada e, em seguida a sua aplicação, um processo de avaliação pode ser iniciado com o objetivo de identificar os efeitos da solução proposta sobre o problema de entrada.

Se a qualidade da solução não for satisfatória, um processo de reparo pode ser iniciado de forma automática (retornando à fase de busca e recuperação, com a possibilidade de um novo ajuste de situação e de índices) ou realizando reparos através de indicações do usuário.

2.2.3.5 APRENDIZADO E ESQUECIMENTO

A última fase do processo trata do aprendizado do sistema, cuja importância é comentada por Kolodner:

... novos casos dão ao raciocinador, contextos familiares adicionais que ajudarão a resolver problemas ou a avaliar situações. Um raciocinador cujos casos cobrem uma grande parte do domínio de conhecimento, será melhor do que um que cubra uma menor parte. Um, cujos casos cubram exemplos falhos e exemplos de sucesso será melhor do que aquele que cobre apenas exemplos de sucesso. Um bom conjunto de índices permite que os casos sejam lembrados nas horas apropriadas e permite que se faça generalizações ... (KOLODNER, 1993, p. 07).

Assim, o aprendizado em um sistema de RBC é dividido em acumulação de casos e associação de índices, proporcionando uma boa cobertura do domínio e permitindo generalizações (KOLODNER, 1993).

O aprendizado de um sistema deve ocorrer de forma ordenada para não tornar a

base de casos algo difícil de ser manipulado. A inclusão de novos casos e associação de índices deve ocorrer de forma que o sistema possa raciocinar sobre eles. Melhorias na medida de similaridade, assim como nas regras de adaptação também podem ajudar no melhoramento da performance do sistema.

A base de casos tende a crescer com o passar do tempo, portanto pode ser necessário a inclusão de um processo de esquecimento para controlar esse crescimento. Este processo pode ser guiado através de funções, regras e heurísticas dependentes do domínio de atuação do sistema.

2.2.4 VANTAGENS E DESVANTAGENS

RBC fornece algumas vantagens e desvantagens (KOLODNER, 1993) ao raciocinador.

Vantagens:

- Permite que o raciocinador proponha soluções rapidamente.
- Permite que o raciocinador proponha soluções em domínio que não entende completamente.
- Fornece ao raciocinador, melhor de avaliar as soluções quando não há método algorítmico para tal.
- Casos são úteis para interpretar conceitos mal definidos e abrangentes.
- Permite que o raciocinador evite potenciais problemas através da lembrança de experiências passadas, alertando-o para executar ações de prevenção.
- Ajuda o raciocinador a focalizar sua atenção em partes importantes de um problema, evidenciando suas características mais importantes.

Desvantagens:

- Permite que o raciocinador use os casos passados sem validá-los na nova situação.
- Permite que os casos influenciem muito, o raciocinador, na resolução de um novo problema.
- Quando se trata de raciocinadores com pouca experiência, a técnica permite que eles utilizem um conjunto de casos não apropriados em seu processo de raciocínio.

2.2.5 APLICABILIDADE

Segundo WEBER (1997), para utilizar-se RBC é interessante avaliar se a tarefa para a qual o sistema é proposto, quando executada por um especialista humano, é uma tarefa baseada em casos.

Ainda segundo WEBER (1997), alguns fatores operacionais e quantitativos podem ser questionados, entre os quais destacam-se:

- na resolução do problema em questão, aplica-se o tipo de raciocínio baseado em casos?
- conhecimento dentro do domínio da aplicação é organizável e expansível?
- existem casos disponíveis e estes são modeláveis computacionalmente?
- desenvolvimento de uma métrica de similaridade é possível?

A utilidade de um sistema de RBC pode ser verificada em variadas situações, como mostra KOLODNER (1993):

- é útil para pessoas que possuem muito conhecimento sobre uma determinada tarefa ou domínio por que fornece um meio de armazenar, reusar e raciocinar sobre toda a experiência adquirida durante algum período de tempo;
- é também útil para capacitar as pessoas que não possuem grande experiência em uma tarefa ou domínio, a resolver problemas partindo de experiências de outras pessoas;
- é útil ainda quando o conhecimento existente é incompleto e/ou as evidências são esparsas. Um sistema de RBC pode fazer asserções para preencher um conhecimento incompleto ou perdido, baseando-se em suas experiências passadas.

2.3 SISTEMAS HÍBRIDOS

2.3.1 CONSIDERAÇÕES GERAIS

As técnicas de Inteligência Artificial têm conseguido suprir muitas necessidades na resolução de problemas antes impossíveis ou extremamente complexas de serem satisfeitas com técnicas tradicionais. Contudo, certos problemas complexos não são solucionáveis com a aplicação de apenas uma técnica inteligente e, para resolvê-los, a aplicação de várias técnicas em conjunto tem apresentado ótimos resultados.

A adoção deste comportamento deu origem a uma nova área dentro da

Inteligência Artificial denominada de Sistemas Híbridos (SH) que, segundo GOONATILAKE e KEBBAL (1995) é a combinação e/ou integração de duas ou mais técnicas inteligentes ou a combinação e/ou integração de uma destas, com técnicas tradicionais, de maneira a suprir as limitações individuais de cada uma delas.

Inicialmente, os esforços dentro dos estudos em SH se concentraram na combinação de Sistemas Especialistas e Redes Neurais porém, atualmente, diversas combinações utilizando, além destas duas técnicas, as técnicas de Lógica Difusa, Algoritmos Genéticos e Raciocínio Baseado em Casos e também técnicas convencionais vem tomando uma posição de destaques dentro da teoria de SH (CARVALHO, 1998).

Segundo MEDSKER (1995), o desenvolvimento de SH integra diferentes ciências incluindo biologia, psicologia cognitiva, linguística, e ciência da computação. Os esforços dispendidos neste campo têm contribuído para melhor resolver problemas baseados em conhecimento e em raciocínio de senso comum.

2.3.2 CLASSIFICAÇÕES

Dentro do estudo de SH, uma classificação foi apresentada por Medsker e Bayle em 1992, apud (MEDSKER, 1995, p. 21), dividindo-os em cinco categorias: modelo *stand-alone*, transformacionais, fracamente acoplados, fortemente acoplados e totalmente acoplados, como mostra a Figura 2.3.

Na primeira classe, onde estão inseridos os sistemas híbridos mais simples de serem construídos, duas ou mais técnicas podem ser usadas, em paralelo, para resolver o mesmo problema, contudo, elas são independentes e não interagem entre si. Este tipo de abordagem pode ser usada quando se pretende comprovar a solução apresentada por uma técnica através da aplicação de outra para resolver o mesmo problema. Sua desvantagem é que o SH desenvolvido é de difícil manutenção já que, mudanças na especificação do sistema, por exemplo, deverão ser refletidas na implementação das duas técnicas e, deverão ser feitas ao mesmo tempo para que a mudança não afete a confiabilidade de comparação das soluções. Outra desvantagem é que, os esforços para chegar a solução são dobrados, ou seja, não são transferíveis como ocorre com outros tipos de SH.

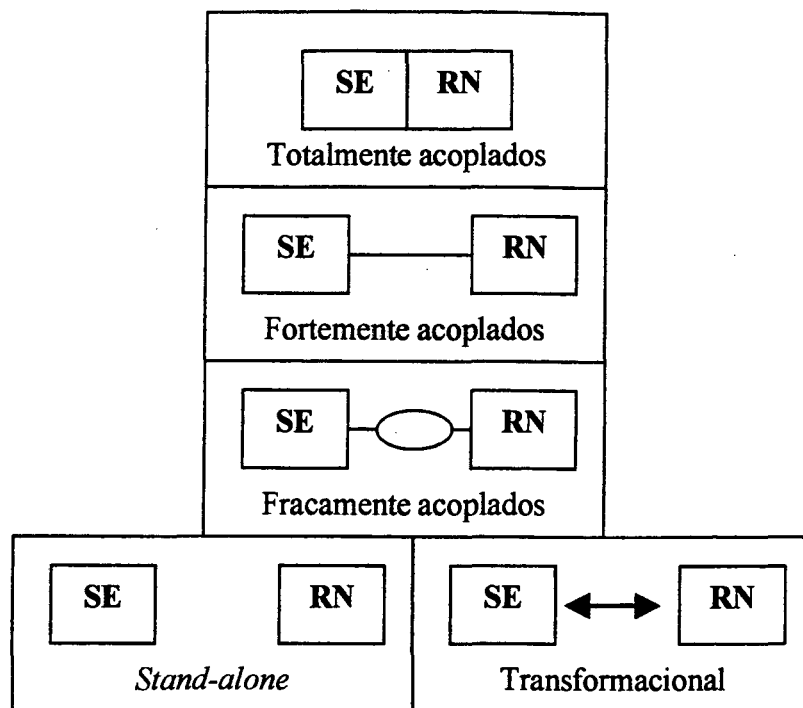


Figura 2.3: Modelos de integração de sistemas inteligentes [Medsker and Bailey, 1992] apud (Medsker, 1995, p.21).

Nos modelos transformacionais, diferentes técnicas estão envolvidas na construção do sistema mas, a execução de cada uma é independente da outra, sendo que, uma técnica deve apenas fornecer dados, diretamente, para a outra, o que caracteriza sua classificação em “transformacional”. Esses sistemas tem um desenvolvimento relativamente simples e sua manutenção pode ser feita de maneira isolada em cada uma das técnicas, porém, modificações significativas podem exigir um nova “transformação”.

A próxima classificação, fracamente acoplados, segundo MESKER (1995), é considerada a primeira forma verdadeira de SH inteligentes. Aqui, a comunicação entre as diferentes técnicas é realizada através de arquivos e pode ser feita em apenas uma direção (de uma técnica para outra) ou em várias direções, permitindo uma interatividade maior e consequentemente, um comportamento de cooperação entre as técnicas. A desvantagem destes sistemas é que, o uso de arquivos para a comunicação leva a um aumento no tempo de operação do sistema.

Sistemas que se comunicam através do uso de informações contidas nas mesmas estrutura de dados estão inseridos dentro da classe dos sistemas fortemente acoplados. Assim como nos sistemas fracamente acoplados, o fluxo de dados pode ocorrer em

qualquer direção. O diferencial é que nos sistemas fortemente acoplados, isto se dá de forma extremamente rápida. Este sistema também podem ser compostos de módulos de diferentes técnicas auxiliando o funcionamento de outros módulos implementados com outras técnicas. O desenvolvimento e manutenção destes tipos de sistemas têm uma complexidade maior do que os sistemas inseridos nas classes descritas anteriormente, devido a dificuldade em construir a interface interna para passagem de dados.

Por fim, a classe dos modelos totalmente integrados compartilham estrutura de dados e representações de conhecimento. Em MEDSKER (1995), encontra-se uma divisão destes quando se trata da interação das técnicas de sistemas especialistas e redes neurais, incluindo:

- sistemas especialistas conexionistas: onde pedaços de conhecimento das ligações pesadas de uma rede neural está diretamente relacionada com nós simbólicos;
- utilização de nós de entrada/saída: os fatos para um sistema especialista são obtidos diretamente dos nós de uma rede neural;
- conectividade subsimbólica para simbólica: os nós intermediários ou os padrões de ativação de uma rede neural são transformados em símbolos através de um sistema especialista ;
- mecanismos de controle integrados: as diferentes técnicas são utilizadas para controlar ou auxiliar o funcionamento de outras, porém, diferentemente dos sistemas fortemente acoplados, aqui as diferentes técnicas não são módulos independentes e não podem ser construídas ou mantidas isoladamente.

Uma segunda classificação é apresentada por GOONATILAKE e KEBBAL (1995), dividindo os SH em três grandes grupos:

- *function-replacing hybrids*: a principal função de uma técnica é alocada para outra com o objetivo de alcançar melhores resultados (realidade) ou melhora da performance (velocidade);
- *intercommunicating hybrids*: vários módulos inteligentes com processamento independentes, executam funções distintas para gerar soluções e trocam informações entre si;
- *polymorphic hybrids*: usam uma arquitetura de processamento única para alcançar a funcionalidade de diferentes técnicas de processamento inteligente.

A Figura 2. mostra as combinações que podem ser feitas entre as diferentes técnicas e a aplicabilidade de tais combinações. Como se observa, para trabalhar com informações lineares, redes neurais, algoritmos genéticos e técnicas matemáticas são preferíveis e para conhecimento formal, sistemas especialistas, sistemas difusos, raciocínio baseado em casos e também, complexas formulações matemáticas podem ser utilizados.

Quanto ao ambiente onde o sistema agirá, pode-se dizer que em ambientes adaptativos e dinâmicos, algoritmos genéticos, sistemas difusos, redes neurais e, de certa forma, raciocínio baseado em casos podem se mostrar mais eficientes enquanto que para ambientes estáticos, o uso de sistemas especialistas, técnicas matemáticas e raciocínio baseado em casos são suficientes.

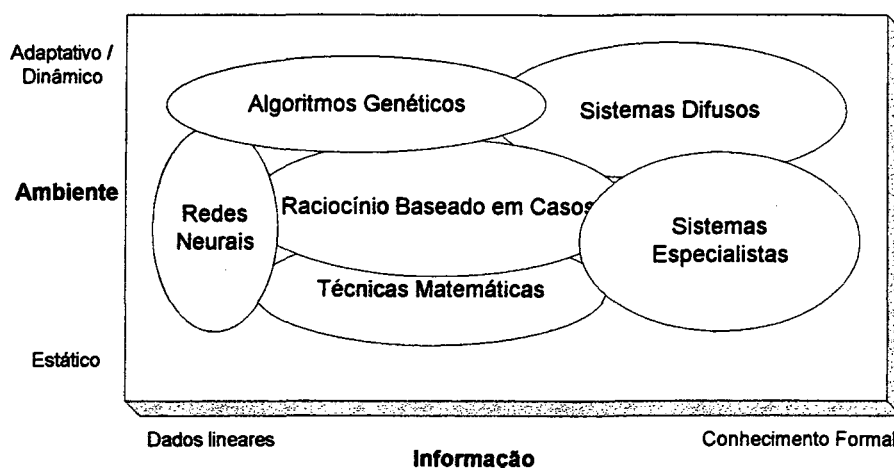


Figura 2.4: Visão geral da integração de tecnologias convencionais e inteligentes (MEDSKER, 1995, p. 229).

2.4 APLICAÇÕES E EXEMPLOS

De acordo com REATEGUI e CAMPBELL apud (CARVALHO, 1998), existem quatro abordagens as quais generalizam os diferentes paradigmas utilizados para integrar, RBC com outros métodos de raciocínio:

- Controle Central: onde RBC e outra técnica são controlados por um dispositivo central;
- Controle Distribuído: onde o controle é dividido entre as duas técnicas;
- Fortemente RBC: quando o sistema RBC exerce um controle maior;
- Fracamente RBC: quando a outra técnica exerce um controle maior.

Uma outra forma de integração se dá pelo projeto de uma arquitetura inteligente, combinando características das várias técnicas envolvidas.

Entre os trabalhos integrando RBC e outras técnicas, estão os que integram Redes Neuras e RBC. Nestes trabalhos, Redes Neurais podem ser utilizadas para indexar e recuperar casos procurando padrões de similaridade entre eles, ou para a aquisição de casos. Neste último, a rede pode ser treinada com novos casos e a partir daí, regras podem ser extraídas da rede para construção de uma base. Uma outra alternativa é o uso de redes como fornecedoras de conhecimento para o processo de raciocínio do RBC.

Em Carvalho (1998), uma série de trabalhos envolvendo essas duas técnicas são apresentadas. Entre eles têm-se:

- sistema de tratamento de esgoto apresentado por Krovvidy e Wee, um sistema RBC é utilizado para fornecer um estado inicial de uma solução para uma rede de Hopfield (PANDYA, 1996) que é então utilizada para sugerir uma sequência ótima de processos para o tratamento de esgoto. Quando uma nova sequência de processos é necessária para o tratamento de determinado esgoto, as concentrações dos elementos contaminadores deste esgoto são utilizados para indexar um tratamento que teve sucesso, utilizando concentrações semelhantes;
- os trabalhos de Malek propõem um mecanismo de indexação neural para RBC que utiliza uma memória hierárquica de casos. O nível mais baixo contém casos organizados dentro de grupos de casos semelhantes e o nível mais alto contém protótipos, cada um representando um grupo de casos;
- no trabalho de Reategui, Campbell e Borghetti, padrões de similaridade entre casos de um sistema de RBC são aprendidos por um Rede Neural, e são utilizados para avaliação psicológica de candidatos a transplante de coração. O conhecimento implícito armazenado na rede é extraído e interpretado simbolicamente e é utilizado para guiar o processo de raciocínio de RBC;
- no trabalho Reategui e Campbell, um sistema de RBC utiliza casos específicos para determinar quando uma transação de cartão de crédito pode ser admitida ou recusada, e a rede neural reconhece padrões gerais de comportamento para o uso incorreto ou corretos de cartões de crédito. Este conhecimento é utilizado na classificações das transações;

- no trabalho de Milaré e Carvalho, foi utilizada uma rede neural ART 1 (PANDYA, 1996) para indexar um sistema de RBC para o domínio da culinária.

Em SUN (1995), uma arquitetura conexionista com representação dual, unifica padrões comuns de raciocínio de senso comum através da incorporação de raciocínio baseado em regras e raciocínio baseado em similaridade.

Uma metodologia para resolver problemas com características desconhecidas e tomar decisões na presença deste desconhecimento é apresentada em (CHENG e WEI). Esta metodologia visa recuperar velhos casos em um RBC como um problema de decisão, onde cada caso da base de casos fornece uma solução alternativa e uma predição de resolução para o problema. Quando a resolução de problemas baseada em casos encontra incerteza, a metodologia aplica teoria da decisão para avaliar cada casos em termos de atributos que são significantes para o problema, tal que os casos mais desejáveis possam ser selecionado.

O trabalho apresentado em (SHEN, LUI e DING) introduz o conceito de raciocínio baseado em casos aproximado, utilizando RBC, lógica difusa e redes neurais. Uma implementação mapeia a implicação entre premissas e conseqüências de regras difusas para pesos de uma unidade de rede neural correspondente e um raciocínio baseado em casos aproximado em redes neurais pode ser realizado.

RBC e Raciocínio Baseado em Regras (RBR) são combinados em um Sistema Baseado em Conhecimento, descrito em XU (1996). Este sistema procura descobrir comportamentos com riscos de AIDS, fazendo uso do conhecimento heurístico e casos prévios para melhorar a acuracidade de um RBR através de um RBC. Este sistema usa também, regras para gerar uma justificativa para um objetio alvo e usa casos para manusear exceções para regras, causadas pelo domínio complexo. O sistema, primeiro executa o RBR que é complementado pelo RBC.

Em MEDSKER (1995), o autor lista alguns exemplos de sistemas que utilizam RBC com outras técnicas inteligentes, entre eles estão:

- Kowalski, pesquisou questões de representação de conhecimento para o processo de raciocínio jurídico. Seu trabalho demonstrou que abordagens estruturadas de representação de conhecimento disponíveis em *shells* comercialmente baratas podem ser usadas para construir sistemas legais baseados em casos;

- Vo e Macchion, usaram RBC para entender domínios de conhecimento para sistemas especialistas suportarem o desenvolvimento de satélites. O sistema híbrido era usado, por exemplo, para sugerir causas de anomalias quando um incidente acontecia, baseado em características relevantes de incidentes similares passados;
- Karamaizis e Feyock, estudaram o uso de uma abordagem híbrida de RBC e sistemas especialistas para diagnosticar e prognosticar falhas em voo para subsistemas de avião;
- Gui, discute uma metodologia híbrida para modelar mudanças de produtos em montadores de produtos. No trabalho estuda-se questões de projetos de uma metodologia orientada a objetos que usa lógica difusa para avaliar alternativas de projetos que são gerados por sistemas de RBC;
- Ajenstat *et al*, tem usado as tecnologias de RBC, Redes Neurais e Sistemas Especialistas para melhorar a população de gado leiteiro. As três técnicas são usadas para melhorar a classificação. A redundância dos três métodos fornece reforço na validação da decisão quando seus resultados são similares e sugerem análises mais profundas quando seus resultados são contraditórios;
- Myllyncki e Tirri, usam Redes Neurais para recuperar casos para o sistema de RBC melhorando sua eficiência;
- Louis, McGraw e Wyckoff, discutem o uso de RBC para explicar os resultados de aplicações de Algoritmos Genéticos. Neste trabalho, ferramentas de análise e de aquisição de conhecimento usadas em RBC são usadas para extrair informação do processo do Algoritmo Genético;
- Ramsy e Grefenstette, propõe um método baseado em casos para inicializar o Algoritmo Genético. A técnica de RBC é utilizada para tirar vantagem do conhecimento de estratégias anteriores e direcionar a busca do Algoritmo Genético.

3 PLANEJAMENTO DE ROTAS

3.1 CONSIDERAÇÕES GERAIS

Com o desenvolvimento das cidades e da tecnologia de transportes, as malhas viárias urbanas, as redes de rodovias, ferrovias ou até mesmo o espaço de navegação marítima e aérea vêm se tornando cada vez mais complexos.

O número de veículos, a existência de vias de mão única, a quantidade de semáforos, entre outros são exemplos de variáveis que vêm contribuindo para o agravamento do fluxo nos sistemas viários.

Neste contexto, é extremamente complicado para uma pessoa, encontrar uma boa rota, principalmente quando ela não está totalmente familiarizada com o ambiente no qual essa rota está sendo estabelecida. Segundo LIU (1996), a ineficiência de rotas encontradas por humanos resulta em uma grande quantidade de viagens perdidas, ocasionando uma perda de 45 bilhões de dólares por ano nos Estados Unidos.

De forma genérica, uma rota pode ser definida como um caminho em um grafo. Segundo RABUSKE (1992), um grafo G é definido como sendo um par ordenado (V, E) , onde V é um conjunto e E uma relação binária sobre V . Os elementos de V são denominados de *vértices* ou *pontos* ou *nós*, e os pares ordenados de E são denominados de *arestas* ou *linhas* ou *arcos* do grafo. Ainda segundo RABUSKE (1992) caminho é qualquer seqüência de arestas onde o vértice final de uma aresta é o vértice inicial da próxima.

Em GOLDEN (1988b), encontra-se uma definição para o problema fundamental de planejamento de rotas para navios da guarda costeira, a qual deve ser modificada quando o problema é estendido para diferentes restrições e diferentes domínios. Esta definição diz que: o problema fundamental de planejamento de rotas assume uma única base e um único veículo. N bóias devem ser visitadas dentro de um período de tempo e com um gasto mínimo de esforço (custo). O problema tem diversas entradas incluindo: a posição de cada bóia, recursos necessários para visitar cada bóia, tempo necessário para visitar cada bóia, território no qual o navio pode navegar, velocidade do navio, etc...

Generalizar esta definição não é uma tarefa fácil visto a grande diversidade de situações onde um planejamento de rotas pode ser aplicado. Entretanto, para a aplicação deste trabalho a seguinte definição é adotada:

“O problema de planejamento de rotas assume uma única base e um único veículo. Uma aresta ou diversas arestas consecutivas de um grafo (formando um caminho) devem ser percorridas, sendo que a escolha destas arestas deve satisfazer a restrições definidas pelo ambiente onde a rota deve ser estabelecida e a restrições impostas pelo requisitante do plano de rota.”

Este capítulo é dividido em seções sendo que, na primeira seção apresenta algumas divisões e definições inerentes ao contexto de planejamento de rotas. A seção 3.3 mostra como abordagens tradicionais são utilizadas para resolver este problema. Alguns fatores importantes para o estabelecimento de rotas são discutidos na seção 3.4 e por fim, na seção 3.5, alguns exemplos de aplicação são apresentados.

3.2 DIVISÃO E DEFINIÇÕES

O problema de planejamento de rotas pode ser dividido em duas grandes categorias: inter e intra cidades. Geralmente, os problemas da segunda categoria envolvem distribuição de mercadorias/pessoas/recursos dentro de malhas viárias urbanas e requerem um detalhamento maior de informações quando comparados com os problemas da primeira categoria. Os problemas da primeira categoria, geralmente, atuam em bases de dados geográficas estaduais e federais com um nível de detalhes mais alto.

Segundo PSARAFITS (1988) planejamento de rotas pode ser denominado como roteamento de veículos. PSARAFITS (1988) define duas classes de problemas de roteamento de veículos: o roteamento estático e o dinâmico. No problema de roteamento de veículos estático os dados de entrada não mudam durante a execução do algoritmo que o resolve ou durante a execução da rota. Já no problema de roteamento de veículos dinâmico os dados de entrada podem mudar ou serem atualizados durante a execução do algoritmo ou durante a execução da rota. Sendo assim, o problema de roteamento de veículos dinâmico deve considerar alguns pontos a mais do que o roteamento estático, listados e discutidos por (PSARAFITS, 1988, p. 225), e entre eles estão:

- a dimensão do tempo é essencial;
- a informação futura pode ser imprecisa;
- eventos próximos, no tempo, são mais importantes;
- mecanismos de atualização da informação são essenciais;
- re-sequenciamento e re-atribuição de decisões deve ser garantida;
- computação veloz é necessária.

Como heurísticas para resolver estes problemas, CHISTOFIDES (1985) apresenta duas alternativas: construção de trechos em paralelo, dividindo o espaço de busca da solução; utilização de métodos de duas fases onde a primeira fase consiste em *clusterizar* regiões para cada veículo (quando o problema envolve uma frota de veículos) e na segunda fase, formular as rotas para cada um dos *clusters*.

Em STATHOPOULOS et al. (1994), os autores definem o planeamento de rotas como o processo de tomada de decisão sobre escolha de rotas antes de sua execução, onde o motorista pode usar sua experiência e conhecimento sobre o ambiente (malhas viárias e condições de tráfego) para avaliar as alternativas disponíveis e selecionar a rota que melhor satisfaz seus objetivos.

O planeamento de rotas pode ser abordado de diversas formas. Entre elas podemos citar:

- problema do caixeiro viajante (LAWLER, 1985) ;
- problema do carteiro chinês (LEE, 1994);
- problema da tour do cavaleiro (HOFFMAN, 1985);
- problema do mensageiro (HOFFMAN, 1985);
- problemas de entrega e/ou coleta com e sem janelas de tempo (GOLDEN, 1988a);
- problemas de planeamento de rotas (HAIGH, 1995).

O problema do caixeiro viajante é um problema clássico dentro da teoria de problemas de otimização e transportes. Para entendê-lo, imagine que um caixeiro, iniciando na cidade onde mora, precisa visitar exatamente uma vez cada cidade pertencente a uma lista e retornar à sua casa (ciclo hamiltoniano⁴). É desejável para ele selecionar a ordem na qual ele visitará as cidade de forma que a distância total

⁴ Ciclo Hamiltoniano: caminho formado dentro de um grafo, de forma que todos os nós do grafo sejam incluídos.

percorrida seja a mínima. Para resolver este problema, é conhecido, para cada par de cidades, a distância entre uma e outra, e é sabido que o caixeiro só deve visitar uma cidade de cada vez e que ele não pode estar ao mesmo tempo em duas cidades.

Existem algumas variações deste problema como por exemplo:

- com repetição de cidades: onde as cidades devem ser visitadas pelo menos uma vez, ao invés de exatamente uma vez;
- com múltiplos caixeiros: onde cada uma das cidades deve ser visitada por um dos m caixeiros envolvidos;
- com pre-determinação de arcos: onde a rota formulada deve conter determinados arcos em direções também determinadas.

A importância do problema do caixeiro viajante não está somente em sua aplicabilidade para problemas reais mas, na sua complexidade como um problema de otimização combinatorial, já que o aumento de uma cidade na lista de cidades causa um aumento exponencial nas possibilidades de solução.

O problema do carteiro chinês é outro clássico desta classe de problemas. Neste problema, um carteiro precisa percorrer todas as ruas de uma área, iniciando e terminando no mesmo ponto, formando um caminho que tenha o menor comprimento possível (ciclo euleriano⁵).

Um outro problema relacionado é o problema da tour do cavaleiro, onde um ciclo hamiltoniano para um grafo cujos vértices são os 64 quadrados de um tabuleiro de xadrez com dois vértices adjacentes se e somente se esse vértices correspondam a quadrados adjacentes no tabuleiro.

Um precursor do problema do caixeiro viajante, é o problema do mensageiro dado por Menger em 1930. Este problema surgiu em conexão com uma definição alternativa de comprimento de curva que Menger propôs:

O comprimento de uma curva pode ser definido como o menor limite superior do conjunto de todos os números que poderiam ser obtidos por pegar cada conjunto finito de pontos da curva e determinar o comprimento do menor grafo poligonal juntando todos os pontos, Menger (1930) apud (HOFFMAN, WOLFE, 1985).

⁵ Ciclo Euleriano: caminho formado dentro de um grafo, de forma que todas as arestas do grafo sejam incluídas.

Este problema foi chamado de problema do mensageiro e se resume a encontrar o menor caminho juntando todos os pontos de um conjunto finito de pontos, cujas distâncias são conhecidas (aos pares) sem, no entanto, formar um ciclo.

Os problemas de entrega e/ou coleta com e sem janelas de tempo⁶, são aqueles onde, no estabelecimento da rota deve ser levado em consideração tanto a natureza do problema de encontrar uma rota quanto algumas características inerentes ao ambiente para o qual o roteamento está sendo elaborado. Dentre essas características podemos citar:

- tamanho da frota de veículos envolvidos;
- número de motoristas;
- média de paradas;
- rota inter e intra cidades;
- integração de rotas com outras atividades;
- tamanho das janelas de tempo nos problemas que as consideram.

Os fatores listados acima, fornece uma característica de escalonamento ao problema, que pode ser trabalhada de maneira isolada ou de maneira conjunta ao estabelecimento da rota.

3.3 ABORDAGENS TRADICIONAIS

Diferentes formas para resolver o problema de planejamento de rotas vem surgindo dentro de várias áreas, entre elas, Pesquisa Operacional e Grafos.

A Matemática é uma disciplina sempre presente dentro de formulações que pretendem resolver o problema de planejamento de rotas ou problemas relacionados. Nestes problemas, algumas questões são essenciais e aparecem na maioria das implementações, sendo formuladas através de cálculos matemáticos e são elas: cálculo das distâncias e cálculo do tempo de viagem.

O cálculo das distâncias pode ser feito com base em coordenadas (x,y) de um plano ou usando informações de um banco de dados que relatam as distâncias verdadeiras. Todavia, estas bases de dados detalhadas não são disponibilizadas facilmente e podem aumentar consideravelmente o esforço computacional.

⁶ Janelas de tempo são restrições de tempo impostas pelo problema de forma que a solução para este, quando executada, não deve ultrapassar um limite de tempo.

O tempo de viagem pode ser obtido através da conversão das distâncias usando uma função de velocidade média, sendo que a determinação da velocidade média deve levar em conta fatores como: não linearidade da velocidade média, influência de horários de *rush* e congestionamentos.

A Teoria dos Grafos, campo inerente também à Matemática, proporciona uma forma de visualização do espaço do problema, através do conjunto de pontos (nós) conectados por linhas (arestas) ou flechas (arcos). Em AHO (1995) encontra-se os principais tópicos relacionados a esta teoria, inclusive uma descrição e comparação sobre o Algoritmo de Dijkstra e o Algoritmo de Floyd, para encontrar caminhos mínimos. Em CORMEN (1990), também encontra-se uma análise sobre os algoritmos para encontrar caminhos mínimos em grafos, enfatizando o Algoritmo de Dijkstra e o Algoritmo de Bellman-Ford.

Ainda dentro da Matemática, outras abordagens são utilizadas para resolver problemas relacionados. Entre elas estão Redes Estocásticas, Processos de Decisão de Markov e Programação Estocástica. A aplicação destas abordagens em problemas de roteamento são encontradas em POWELL (1988).

A Teoria da Probabilidade é usada para resolver variações probabilísticas de problema clássicos como: roteamento de veículos probabilístico e caixeiro viajante probabilístico. Nesta abordagem, apresentada por JAILLET (1988), subconjuntos de clientes ou cidades são visitados em cada instância do problema, sendo que a determinação deste subconjunto é guiada por alguma lei probabilística.

A Programação Linear também apresenta soluções para problemas desta natureza, modelando o problema de forma a resolvê-lo encontrando o mínimo de uma função linear. No caso do problema do caixeiro viajante, a função linear está relacionada a um poliedro descrito por um sistema de equações lineares e inequações de variáveis negativas. Desde que os pontos para os quais o mínimo é obtido deve incluir um vértice do poliedro, e o número de vértices é finito, Programação Linear pode ser usada, em princípio, como uma ferramenta de otimização combinatorial.

Os modelos de programação linear abrangem uma grande variedade de problemas específicos dentro dos problemas de transporte. O método genérico simplex (HILLIER, 1995) é um algoritmo poderoso que pode resolver surpreendentemente, muitas versões destes problemas. Contudo, ainda segundo HILLIER (1995), alguns

problemas são resolvidos de forma mais eficiente com algoritmos “streamlined”, que exploram estruturas especiais de representação, diminuindo consideravelmente o esforço computacional requerido.

Para versões geométricas problemas que envolvem um ciclo hamiltoniano (onde os nós dos grafos são modelados em um espaço métrico e os pesos dos arcos são computados com medidas espaciais), pode-se utilizar a Teoria de Desigualdade Triangular onde considera-se que a menor distância entre dois pontos é uma linha reta. Dentro desta abordagem, alguns algoritmos clássicos podem ser utilizados (JOHNSON, 1985b, p. 15):

- algoritmo do vizinho mais próximo: que basicamente, escolhe um ponto para iniciar a rota e escolhe os próximos, tomando sempre aquele que se encontra mais próximo do último que entrou na rota. Um problema deste algoritmo é que, embora todas as arestas sejam as menores, a aresta final que une o último ponto ao primeiro, pode ser muito longa;
- árvore de espalhamento mínimo: uma árvore de espalhamento mínimo é criada a partir do grafo que modela o problema e somente é permitido utilizar as arestas desta árvore para compor a rota. Uma abordagem utilizada é iniciar a rota em uma folha da árvore (vértice do grafo) e aplicar a seguinte estratégia – se existir qualquer aresta saindo do vértice atual, que ainda não foi percorrida, siga-a para um novo vértice; se todas as arestas já foram percorridas, retorne ao longo das arestas através dos vértices que já foram visitados e repita o procedimento; pare quando retornar ao vértice inicial. Uma técnica para encontrar a árvore de espalhamento mínimo é encontrada em AHO (1995) e CORMEN (1990);
- algoritmo de Christofides: este algoritmo constrói primeiramente um caminho euleriano a partir de uma árvore de espalhamento mínimo modificada e depois utiliza a técnica de “shortcut” (JOHNSON, 1985b) para transformá-la em um caminho hamiltoniano.

Analisando sob o prisma da Matemática, nota-se que os algoritmos criados atualmente para resolver os problemas relacionados ao problema de planejamento de rotas, não diferem, de forma acentuada, dos algoritmos apresentados no início dos trabalhos na área, para a resolução do clássico problema do caixeiro viajante.

O que tem diferenciado, as atuais resoluções das anteriores, é a forte

preocupação com a modelagem de características do mundo real, onde a rota deve ser executada, utilizando heurísticas para criar soluções possíveis de serem executadas, atendendo aos requisitos de praticidade e conforto.

Mais recentemente, o problema de encontrar rotas vem sendo estudado no sentido de gerar rotas satisfatórias para indivíduos com diferentes perfis, onde a preocupação com a geração de rotas de caminho mínimo resolve apenas parte do problema. O custo envolvido na execução desta rota é fator determinante dentro da satisfação do indivíduo e, o custo pode envolver desde tempo, desgaste do meio de transporte até desgaste das pessoas envolvidas na execução desta rota.

Neste sentido, faz-se necessário o desenvolvimento de sistemas que sejam capazes de considerar, eficientemente, os diversos fatores que podem estar envolvidos na satisfação do indivíduo, para a geração e otimização de rotas. Para isto, entretanto, as técnicas de Programação Matemática se apresentam pouco eficientes. Em particular, este problema apresenta características que dificultam o uso destas técnicas, já que propriedades dinâmicas, incerteza, “incompletude”, e dificuldade de avaliação da qualidade da solução são difíceis de serem modeladas matematicamente.

O que faz um motorista escolher entre uma ou outra rota são características altamente subjetivas e que variam de um motorista para outro. A tomada de decisão dentro deste domínio é praticamente baseada em aspirações e, qualquer procedimento que tenha o objetivo de automatizar este processo, deve incorporar estas aspirações (STATHOPOULOS, 1994).

Um fator que ajuda a alcançar este objetivo é utilizar o conhecimento existente sobre o ambiente onde a rota será gerada e também utilizar a experiência adquirida através da geração e execução de rotas. Assim, as técnicas de Inteligência Artificial que exploram o uso de conhecimento (LIU, 1996) e o aprendizado com as experiências passadas (LIU, 1996) e (HAIGH, 1995), se mostram extremamente eficientes para auxiliar na resolução deste tipo de problema.

3.4 FATORES IMPORTANTES NO PLANEJAMENTO DE ROTAS

Para conhecer o que é realmente importante para a tomada de decisão de um motorista quando está diante de uma situação de planejamento de rotas, STATHOPOULOS et al. (1994) realizaram um trabalho onde alguns testes foram feitos

com a participação de diferentes motoristas. Como resultado, alguns critérios foram listados como sendo os mais comuns nas tomadas de decisão, tais como: minimização do tempo, fuga de congestionamentos, minimização da distância, segurança, busca por ruas principais e objetividade e familiaridade com o trajeto.

GÄRLING (1994), discute alguns pontos relevantes para a tomada de decisão na escolha de rotas:

- interdependência de decisões: as decisões podem ser modeladas pela expansão de um conjunto de escolhas de forma a incluir todas as possíveis opções de combinação, objetivando a formação de planos que envolvem alguns pontos defendidos por diferentes autores, tais como a especificação de condições sob as quais as ações devem ser tomadas, a formação hierárquica de planos, iniciando por planos esquemáticos e caminhando para planos mais refinados, a especificação dos mecanismos sob os quais os planos foram gerados;
- aquisição, representação e uso da informação: a escolha por uma ou outra opção deve ser modelada respeitando a maneira como as pessoas adquirem e representam a informação em sua memória. Por exemplo: a informação espacial, representada por um mapa, é com certeza uma forma muito fácil de ser assimilada e aceita por uma pessoa. Outra questão interessante é que toda informação sobre algo, pode influenciar a escolhas das pessoas sendo assim, informações que mostrem riscos ou incerteza não devem ser deixadas de lado;
- regras de decisão: Payne et al. (1993) apud (GÄRLING, 1994) propõe que as pessoas procuram uma visão construtivista de tomada de decisão a qual captura o conhecimento acumulado existente sobre o assunto. Assim, as pessoas usam diferentes regras de decisão heurísticas procurando ajustar a decisão às restrições existentes.
- motivos sociais e egoístas: algumas pesquisas revelam que a tomada de decisão por parte das pessoas, tende a ser benéfica para interesses sociais, sobrepondo qualquer interesse mais egoísta. Percebeu-se também, nestes estudos que, uma alternativa que beneficia interesses pessoais, quando escolhidas por muitas pessoas, se tornam a pior escolha dentro dos critérios egoístas e que as pessoas entendem esta problemática.
- implementação de escolhas: nem sempre o comportamento das pessoas é fiel

ao plano escolhido visto que as preferências mudam com o tempo e, imprevistos no mundo real são constantes.

Uma reflexão sobre todos estes pontos se faz necessária para o desenvolvimento de qualquer trabalho que envolva uma tomada de decisão, principalmente, quando esta decisão está intimamente ligada com a satisfação do ser humano.

3.5 EXEMPLOS DE APLICAÇÃO

O problema de coleta de lixo em uma cidade (BALL, 1988), foi otimizado através de um sistema automatizado e foi uma das primeira aplicações dos algoritmos criados para resolver problemas de planejamento de rotas. Neste problema, uma questão interessante a ser considerada, no planejamento da rota, é a flexibilidade relacionada aos dias em que a coleta deve ser feita e em quais localidades, de forma a assegurar que a quantidade de lixo gerada diariamente não cresça demasiadamente.

Sistemas de distribuição de produtos (BALL, 1988), também automatizados, caracterizam um problema de natureza um pouco diferente, devido a existência de um local de armazenamento dos produtos, localizado de forma estratégica, de onde frotas de veículos são despachados diariamente. Uma das preocupações primordiais neste tipo de problema é evitar que o estoque do armazém diminua tanto a ponto de influenciar negativamente na satisfação do cliente. Existem duas variações para esta aplicação, sendo que cada uma delas exige uma modelagem diferente:

- quando se conhece a quantidade que deve ser entregue para cada consumidor;
- quando não se conhece a quantidade que deve ser entregue para cada consumidor e trabalha-se com estimativas.

Uma outra aplicação foi modelada para projetos de computadores e outros sistemas digitais (JOHNSON, 1985a). Um sistema digital consiste de um número de módulos e vários pinos localizados em cada módulo. A posição física de cada módulo é pré-determinada e um dado conjunto de pinos precisam ser interconectados por fios. Em vista das possíveis mudanças futuras ou correções e do pequeno tamanho de um pino, para a maioria, apenas dois fios devem ser colocados em cada pino. Para evitar cruzamentos e melhorar a transmissão nos fios, o total do comprimento dos fios deve ser minimizado.

Sistemas de Informações Geográficas contendo informações detalhadas sobre redes de estradas e localização do usuário têm sido bastante utilizados para implementação de sistemas para resolução de problemas de planejamento de rotas. Um sistema que utiliza esta tecnologia, apresentado em ASSAD (1988), é o sistema ETAK, vendido por uma empresa californiana de mesmo nome, especialista em desenvolver dispositivos de rastreamento de veículos.

O sistema ETAK fornece, ao motorista do veículo, a sua posição dentro do espaço geográfico modelado, de forma que o veículo apareça sempre centrado na imagem do mapa mostrado através de um pequeno monitor de bordo, ou seja, conforme o veículo se locomove, a região do mapa mostrada ao motorista é alterada. Uma outra funcionalidade deste sistema é que o motorista tem a possibilidade de ajustar o nível de detalhe da informação exibida.

Para localizar o veículo dentro do espaço geográfico modelado, o sistema, a princípio, utilizava fitas cassetes que contêm a base de dados geográficos para pequenas áreas que são subdivisões da área total coberta pelo sistema, sendo necessária a troca de fitas sempre que o veículo passe de uma sub-área para outra e em versões mais recentes, substituiu-se a fita cassete pelo CD-ROM.

Sobre esta base de dados é aplicado um esquema de localização chamado *Dead Reckoning*, o qual rastreia a posição do veículo na malha viária baseada na informação detalhada sobre a sua movimentação. Sensores de navegação combinam informações sobre a velocidade do veículo e mudanças na sua direção, com os dados do espaço geográfico modelado obtendo a posição do veículo que é transmitida ao sistema navegador através de rádio.

Em HAIGH (1993), HAIGH (1994), HAIGH (1995) e HAIGH (1997), os autores descrevem uma abordagem para resolver o problema de planejamento de rotas através de raciocínio analógico, como uma instância de raciocínio baseado em casos. Esta abordagem consiste na reunião e reuso de experiências sobre rotas, onde características geométricas da malha viária em questão são exploradas no processo de raciocínio analógico.

Neste trabalho, várias rotas armazenadas são recuperadas através de um esquema de índices e uma métrica de similaridade que considera características geométricas descritas por valores contínuos combinada com similaridade simbólica. Essas rotas

formam, coletivamente através de um “merge” e da adição de novos passos onde necessário (adaptação), uma nova rota.

O aprendizado, no método apresentado para planejamento de rotas, ocorre através do “feedback” do usuário, garantindo o aspecto qualitativo na biblioteca de rotas. O sistema construído atua sobre uma malha viária de 18.000 cruzamentos e 25.000 segmentos de rua.

Em LIU (1995), o autor descreve uma abordagem que utiliza conhecimento sobre a malha viária para reduzir o tempo e o espaço requerido computacionalmente na geração de rotas e para assegurar uma solução mais dirigida às necessidades humanas. Liu utiliza uma abordagem baseada em conhecimento incorporada à algoritmos de busca como o Dijkstra ou o A* (RICH, 1993). O conhecimento sobre a malha viária é utilizado para diminuir o espaço de busca e/ou guiar a busca.

Dois tipos de conhecimento são utilizados:

- conhecimento sobre o tipo das ruas (ruas pequenas, ruas grandes e rodovias);
- conhecimento sobre quais ruas grandes e vias rápidas podem particionar a malha viária em sub-malhas, onde o algoritmo atuará.

Nesta abordagem, a malha viária fica reduzida a uma malha composta somente por ruas grandes e cada sub-área é considerada como um ponto, que é também uma pequena malha viária a qual só será considerada pelo algoritmo se esta contiver a origem ou o destino da rota procurada. O sistema gerado, KB-Rfinder atua sobre uma malha viária de 12.600 nós e 30.700 segmentos, com uma malha reduzida para 1100 com 58 sub-malhas.

Em LIU (1996), o autor apresenta uma abordagem que integra o uso de conhecimento sobre a malha viária, casos passados e um algoritmo de busca para encontrar uma rota. Este trabalho melhora a abordagem apresentada em LIU (1995). O uso de casos passados vem de encontro à necessidade de considerar a preferência das pessoas por viajar por ruas/estradas familiares. Usando casos, experiências passadas podem ser recuperadas.

O sistema é usado por usuários individuais e produz soluções personalizadas que consideram diferentes preferências deste usuário através do uso de experiências passadas, as quais estão implícitas no sistema na forma de pesos de conexões (ruas). Para as ruas que agradaram o indivíduo no decorrer de uma experiência os pesos são

diminuídos e, para aquelas que não agradaram os pesos são elevados.

O trabalho desenvolvido por BURMEISTER, HADDADI e MATYLIS (1997), apresenta as diversas aplicações da tecnologia de sistemas multi-agentes⁷ dentro de problemas de tráfego e transporte. Os autores caracterizam problemas de tráfego e de transporte como problemas geograficamente e funcionalmente distribuídos, subsistemas que tem um alto grau de autonomia e que são altamente dinâmicos. Essas características justificam a utilização de agentes inteligentes para uma melhor resolução. Dentre as aplicações apresentadas neste trabalho tem-se:

- gerenciamento de tráfego: planejar, implementar, testar e administrar o tráfego para otimizar a relação entre oferta e demanda de recursos de tráfego sob aspectos ambientais e econômicos;
- guia e controle de tráfego: aumentar a autonomia de sistemas de controle de tráfego e melhorar suas capacidades de interação;
- gerenciamento de recursos: planeja rotas e escalonamentos para descarregar, esperar, carregar, despachar e alocar mercadorias.

Esses autores ainda descrevem um sistema orientado a agentes que faz uma simulação de tráfego procurando melhorar o tráfego através da investigação e avaliação de estratégias de controle alternativas.

Nesta simulação cada veículo e cada motorista é considerado um agente e cada agente conhece informações sobre si mesmo (posição, velocidade, etc), sobre outros agentes que se encontram próximos a ele (posição, velocidade, etc) e sobre o ambiente (comprimento de ruas, sentido etc). Esses agentes percebem, agem e reagem de forma a alcançar uma ou mais metas, como por exemplo, dirigir dentro das leis de trânsito.

⁷ Sistemas multi-agentes são sistema moldelados por uma coleção de agentes, sendo que, por agente entende-se: um sistema computacional que possui autonomia, habilidade de comunicação, capacidade de percepção e reação e iniciativa e são usados conceitualmente ou logicamente usando conceitos que são comuns a seres-humanos (WOOLDRIGE, 1995).

4 PROTEUS

4.1 INTRODUÇÃO

O sistema PROTEUS (Planejador de ROTas para Transporte em Espaços Urbanos e Similares) visa, principalmente, elaborar planos de rotas, sobre ambientes urbanos, agindo sobre uma malha viária. Porém, pode ser estendido para atuar em malhas rodoviárias, ferroviárias ou para qualquer outro espaço (marítimo ou aéreo) que possa ser representado como uma malha. Para elaborar a rota, o sistema deve gerá-la ou recuperá-la dentro de uma base e analisá-la conforme as necessidades do usuário que a está requisitando.

Neste sistema, cada plano de rota é composto pela trajetória que compõe a rota - formada por uma série de segmentos adjacentes dentro da malha viária, escolhidos sob determinadas restrições - e por características que descrevem alguns aspectos de cada plano, como por exemplo: pavimentação dos segmentos e velocidade máxima permitida.

As restrições que influenciam na escolha dos segmentos para formar uma rota são impostas por aspectos inerentes à própria malha viária, tal como a existência de ruas de mão única e por preferências do usuário, tais como preferências por ruas asfaltadas.

O conjunto de preferências específicas de um determinado grupo de usuários forma um perfil de usuário. O sistema PROTEUS aceita diferentes perfis de usuários, desde que estes sejam determinados por características representadas dentro do sistema. E, para cada requisição de rota, o sistema procura satisfazer ao usuário, apresentando uma rota que satisfaça, da melhor maneira possível, as suas preferências.

4.2 ARQUITETURA HÍBRIDA

“... alguns autores afirmam que fazer uma rota envolve a formação de um plano e que isto envolve detalhes de resolução de problemas que têm sido estudados em laboratório de psicologia: alternativas são geradas após uma busca heurística em um espaço de solução, avaliadas de acordo com um critério de seleção, selecionadas e depois implementadas...” (GÄRLING, 1994, p. 119).

O sistema desenvolvido é caracterizado como híbrido por envolver duas técnicas de Inteligência Artificial: Algoritmos Genéticos e Raciocínio Baseado em Casos. Dentro das classes definidas na seção 2.3.2, o sistema pode ser classificado como:

- fracamente acoplado;
- intercommunication hybrids e
- fortemente RBC.

A Figura 4., adaptada de (MEDSKER, 1995, p. 216) mostra, dentro do processo de RBC, onde o Algoritmo Genético se integra. Neste sistema, o Algoritmo Genético age como um meio de alimentação da base de casos.

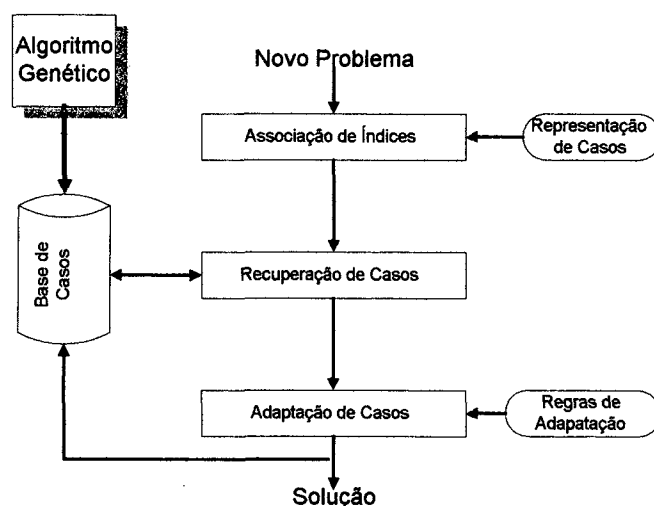


Figura 4.1: O processo de RBC integrado ao uso do Algoritmo Genético, adaptada de (MEDSKER, 1995, p. 216).

A Figura 4.2 apresenta a arquitetura do sistema desenvolvido. Inicialmente o usuário interage com o sistema, fornecendo a origem e o destino desejados e qual é o seu perfil. A partir dessas informações, o sistema inicia sua execução, requisitando ao Módulo RBC, uma busca em sua base de casos permanente – onde as experiências passadas estão armazenadas - para verificar se existe uma rota que resolve o problema do usuário no sentido de levá-lo do ponto de origem ao ponto de destino. Se nenhuma rota for encontrada, o Módulo Genético é ativado.

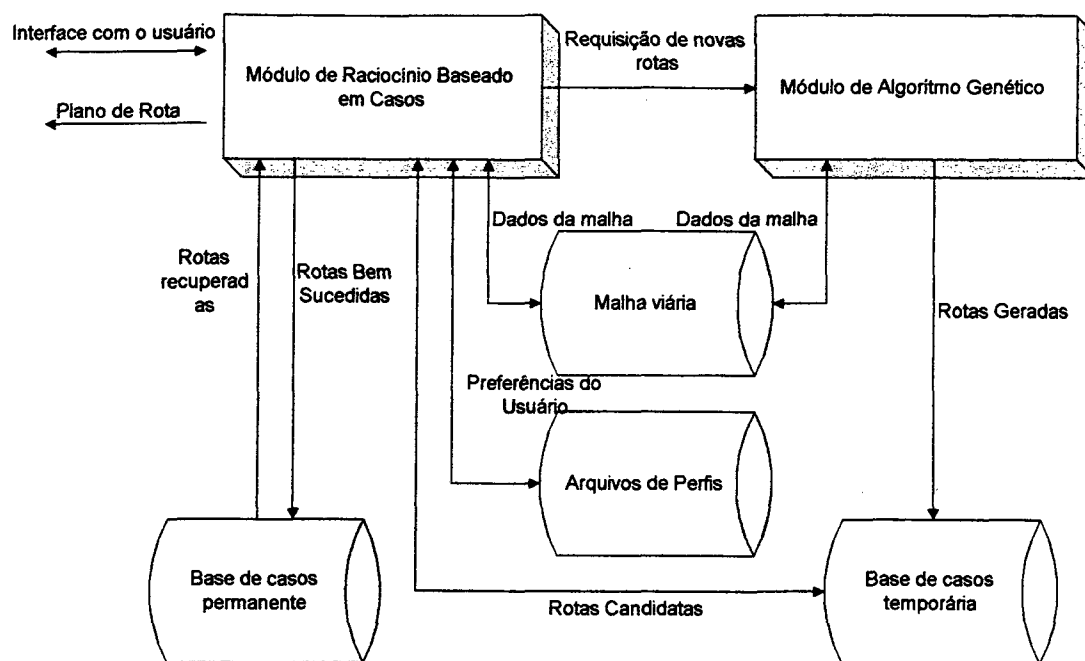


Figura 4.2: Arquitetura do Sistema (PERES, 1997)

O Módulo Genético é responsável pela geração de novas rotas que satisfaçam o problema de entrada, no que diz respeito à origem e destino especificados e à orientação das ruas na malha viária. Essas rotas são armazenadas em uma base de casos temporária para serem analisadas, pelo Módulo RBC, quanto a sua adequabilidade ao perfil do usuário.

Após a obtenção das rotas, independentemente do módulo que as obteve, elas serão analisadas, pelo Módulo RBC, de acordo com as exigências inerentes ao perfil do usuário em questão. Para a realização desta análise, este módulo deve consultar a base de dados onde as preferências relacionadas a cada perfil estão disponíveis e consultar a base onde a malha viária e suas características estão representadas.

Essa análise gera um conjunto de rotas que devem ser ordenadas por nível de similaridade ao problema do usuário (considerando suas preferências) e apresentadas a ele. Se na análise das rotas constatar-se que alguns trechos não satisfazem ao usuário, o sistema decide por adaptar este trecho, tomando-o como uma nova rota a ser gerada ou recuperada, reiniciando o processo.

Após o encontro da solução, esta deve ser executada pelo usuário. Após a execução o usuário pode decidir se ela deve ou não ser armazenada na base de rotas, caracterizando o aprendizado do sistema.

4.3 A OPERAÇÃO DO SISTEMA

Para operação do sistema, o usuário deve informar o ponto inicial (origem) e o ponto final (destino) da rota desejada e a qual perfil ele pertence e requisitar o traçado da rota, como mostra a Figura 4.3:

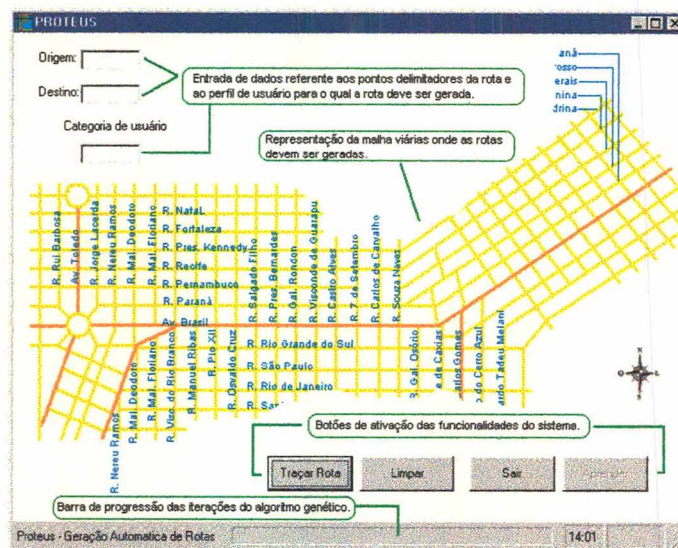


Figura 4.3: Tela inicial do sistema PROTEUS.

A Figura 4.3 ilustra a tela inicial do sistema PROTEUS. As entradas de dados, no canto superior esquerdo da tela, são requeridas para a execução do sistema. A falta de alguma delas dispara uma mensagem com um aviso que as três entradas devem ser fornecidas. Os botões de ativação das funcionalidades do sistema têm as seguintes descrições:

- **Traçar rota:** é o botão que ativa o funcionamento do sistema iniciando a execução do processo de recuperação/geração de rotas, se as entradas estiverem completas, senão, ativa a mensagem de aviso;
- **Limpar:** deve ser clicado quando se deseja gerar/recuperar uma nova rota. Este botão reinicializa as variáveis internas do sistema;
- **Sair:** botão que termina a execução do sistema;
- **Aprender:** este botão só se encontra ativo quando o algoritmo genético gerou algumas rotas ou quando o sistema adaptou uma rota e deve ser ativado se o usuário desejar armazenar as novas rotas na base de rotas, ou seja, este botão ativa o

aprendizado do sistema.

Na parte inferior na tela existe uma barra de progressão que progride de acordo com as iterações do Algoritmo Genético, ilustrando as gerações¹ do algoritmo. Na parte central da tela encontra-se a representação gráfica da malha viária da cidade onde o sistema atua.

4.4 O MÓDULO GENÉTICO

O Módulo Genético² tem como objetivo gerar rotas válidas entre os pontos de origem e destino fornecidos pelo usuário. A estratégia consiste em buscar alternativas válidas, procurando por diferentes rotas de comprimento mínimo.

Este módulo utiliza uma representação para as rotas em códigos binários (cromossomos). Problemas parecidos já foram alvo de trabalhos como o de WHITLEY, STARKWEATHER e SHANER (1991) onde, era permitido combinações entre dois pontos dentro de uma malha totalmente conectada. No problema em foco, cada segmento da rota tem que pertencer a uma caminho realmente possível de ser percorrido. Isto faz com que os operadores usuais de um Algoritmo Genético - mutação e cross-over - sejam inadequados, já que eles podem gerar segmentos não existentes. Neste trabalho, foi desenvolvido uma representação independente de ordem, que os torna aplicáveis em malhas que não são totalmente conectadas. Este esquema permite a modelagem de grafos orientados e garante a validade das rotas.

O Módulo Genético começa pelo estabelecimento randômico de um conjunto de rotas. Cada rota é um conjunto de segmentos orientados, criadas pela adição gradual de segmentos. Esta adição de segmentos é guiada pela distância Euclidiana entre os nós candidatos a pertencer à rota e os ponto de destino. A montagem da rota continua até que o algoritmo encontre o ponto de destino ou até que o número máximo de gerações seja alcançado. O algoritmo genético procede tentando estabelecer várias rotas até que alcance um número máximo fixado (tamanho da população).

Neste trabalho, cada rota é representada como um cromossomo (Figura 4.4) onde:

¹ Cada iteração completa do Algoritmo Genético caracteriza uma geração na evolução da população de cromossomos.

1	2	3	44	5	6	7	8	9	10	...	N
0	*	0	0	*	1	*	0	1	0	...	0

Figura 4.4: Rota representada na forma de um cromossomo.

- um alelo representa um segmento de rua da malha viária;
- cada um dos alelos recebe um dos valores do conjunto $\{0,1,*\}$. O valor '0' significa que o segmento, na iteração corrente, não pertence à rota; o valor '1' indica que o segmento, na iteração corrente, é um candidato a pertencer a rota e o valor '*' indica que o segmento pertence à rota.

Nesta implementação, a *mutação* é um processo de dois passos, composto por (a) *mutação candidata*; e (b) *mutação reset*, descritas a seguir:

A *mutação candidata* é aplicada para cada alelo valorado com '1'. Ela é definida por três operações: (a) um teste de validade, o qual verifica a validade de cada segmento candidato; (b) o cálculo da distância Euclidiana entre todos os candidatos e o ponto de destino; e (c) avaliação, o qual verifica se o alelo será mudado para '*' ou se permanecerá como um candidato, de acordo com o seguinte critério:

- se o candidato pode ser conectado ao conjunto de segmentos atualmente pertencentes à rota (isto é, se é um segmento válido), toma-se sua distância Euclidiana até o ponto de destino (d_c). Toma-se também a distância Euclidiana entre o conjunto de nós atuais e o destino (s_d) (isto é, a distância entre o último segmento pertencente à rota, até então formada, e o ponto de destino). Entre todos os alelos candidatos válidos, toma-se o que está mais próximo do ponto de destino (em relação às distâncias Euclidiana d_c), verifica-se se sua distância d_c é menor que s_d (para garantir que a rota esteja se aproximando do destino) e então, converte-o para o valor '*'.

O operador de *mutação reset* trabalha com cada alelo valorado com '0' ou '1', decidindo randomicamente se o muda ou não para seu valor oposto. Os alelos valorados com '*' permanecem não modificáveis, fixando o esquemata no sentido de assegurar a formação de rotas válidas e a eficiência do algoritmo.

² O Módulo Genético é parte do Trabalho de Pesquisa de Josué Pereira de Castro realizado no Programa de Pós-Graduação em Engenharia de Produção e Sistemas da Universidade Federal de Santa Catarina.

A *avaliação de fitness* ocorre ao final. As rotas encontradas são classificadas de acordo com seu comprimento total (isto é, a soma dos comprimentos dos segmentos pertencentes a cada rota). A saída do algoritmo genético é um conjunto de rotas válidas e otimizadas, onde o critério utilizado é o caminho mínimo, buscado dentro de um grafo orientado e parcialmente conectado. Nenhuma outra característica é levada em consideração pelo Módulo Genético.

Com o intuito de balancear o comprimento com outras características interessantes ao usuário, as respostas do algoritmo genético devem migrar para o Módulo de RBC. Isto é executado através da transformação das saídas do Módulo Genético para o formato de casos armazenados dentro de uma biblioteca de casos temporária.

4.5 O MÓDULO DE RBC

Este módulo, implementado utilizando a técnica de RBC, avalia e, se necessário, adapta as rotas recuperadas da base de casos permanentes ou da base de casos temporária, para atender a uma requisição do usuário. A avaliação é realizada a partir da análise das rotas recuperadas/geradas sob os aspectos que interessam ao perfil do usuário que está utilizando o sistema. A adaptação ocorre sempre que algum trecho da rota não satisfaça às necessidades do usuário, e para isso, o módulo realiza o processo correspondente as fases do RBC novamente, tomando este trecho como sendo uma nova situação. Este módulo é também responsável pelo aprendizado do sistema e desta forma, se destaca por cobrir todo o ciclo do processo de RBC.

4.5.1 REPRESENTAÇÃO DO CONHECIMENTO

As informações disponíveis no sistema consistem em:

- banco de dados contendo o mapa da malha viária da cidade e informações que caracterizam cada um dos perfis de usuários;
- um conjunto de procedimentos que relaciona os dados contidos no banco de dados gerando a informação necessária para avaliação das rotas;
- duas bibliotecas de casos onde estão armazenadas as rotas já executadas por algum usuário e as rotas que são passadas do Módulo Genético para este.

4.5.1.1 A REPRESENTAÇÃO DA MALHA

O mapa é representado em um grafo direcionado, onde os arcos representam os segmentos de rua e os nós representam os cruzamentos. A direção dos arcos indicam o sentido da rua: arcos uni-direcionados representam ruas de mão única e arcos bi-direcionados representam ruas de mão dupla.

Cada nó deste grafo está associado, no banco de dados, a um código que o identifica, a uma coordenada (x,y) que o localiza no mapa e a uma descrição onde se tem informação textual sobre o cruzamento.

Cada arco está associado, no banco de dados a um código que o identifica, e a alguns atributos descritivos. A Tabela 4.1 apresenta o significado de cada atributo descritivo de um segmento de rua. É importante salientar que as características são consideradas, na métrica de similaridade, de acordo com os perfis dos usuários. Diferentes perfis, podem atribuir pesos distintos para cada característica.

Atributo	Significado
Origem	Ponto de origem do segmento
Destino	Ponto de destino do segmento
Mão Única	Valor booleano que Quando marcado significa que a rua tem um único sentido permitido (origem – destino)
Pavimentação	Instanciado com o código 1 significando que o segmento é asfaltado.
Localização	Instanciado com o código 1 significando que o segmento está localizado em uma área residencial.
Velocidade	Instanciado com o valor 2 indicando que a velocidade máxima permitida é 20 km/h.
Comprimento	Instanciado com o valor 1 indicando que o segmento tem 100 m de comprimento
Preferencial na Origem	Valor lógico que quando marcado significa que o segmento tem a preferencial no cruzamento de origem.
Preferencial no Destino	Valor lógico que quando marcado significa que o segmento tem a preferencial no cruzamento de destino.
Semáforo na Origem	Valor lógico que quando marcado significa que o segmento tem um semáforo no cruzamento de origem.
Semáforo no Destino	Valor lógico que quando marcado significa que o segmento tem um semáforo no cruzamento de destino.
Lombada	Valor lógico que quando marcado significa que o Segmento tem uma lombada.

Tabela 4.1: Descrição dos atributos dos segmentos.

Ainda dentro deste banco de dados, outras informações estão associadas as arestas, por meio de relacionamentos:

- um relacionamento com os pontos de forma a representar uma matriz de incidência ;
- um relacionamento a um esquema de representação de características de

sazonalidade (alta temporada de turismo, período letivo) e de horários de *rush* (picos de volume de carros em determinados períodos de tempo);

- um relacionamento com os tipos de veículos que são permitidos trafegar no segmento.

A Figura 4.5 mostra um exemplo de instanciação de parte da base no que diz respeito à representação da malha viária.

4.5.1.2 OS DADOS QUALITATIVOS – CARACTERÍSTICAS DOS PERFIS DE USUÁRIO X DOS SEGMENTOS DE RUA

Outros dados existentes no sistema, são os dados qualitativos, os quais dão suporte a obtenção de informação qualitativa sobre os usuários e suas preferências. Cada categoria de usuário do sistema tem características próprias, definindo perfis que influenciam diretamente na formulação das rotas (Figura 4.6).

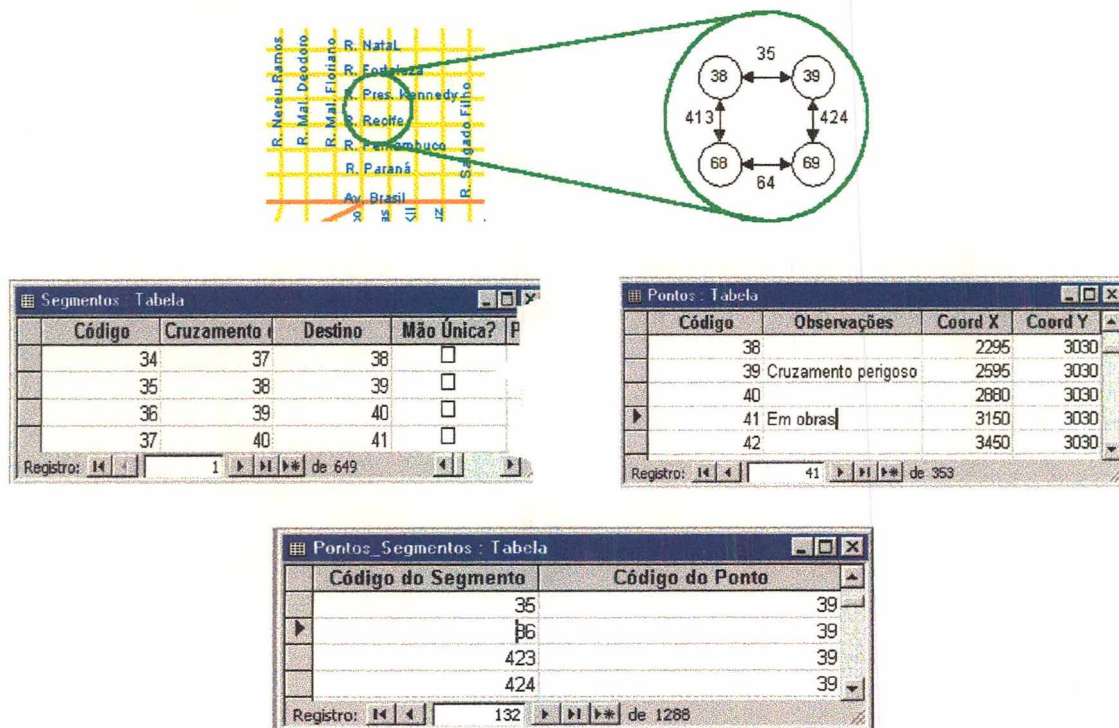


Figura 4.5: Exemplo de representação da malha viária.

Estes perfis estão associados a algumas características descritivas dos segmentos. Essa associação pode ser melhor entendida através da Figura 4.7, onde observa-se, por exemplo, o seguinte relacionamento: a categoria 5, que corresponde aos turistas (Figura 4.6), está associada aos códigos de localização 1, 2, 3 e 4 com as

importâncias 3, 2, 4 e 1 respectivamente. Isto significa que para um turista é preferível passar por uma rua localizada em uma área comercial (código 3, importância 4) do que em uma área residencial (código 1, importância 3) ou ainda pode-se dizer que uma rota que contém muitos segmentos de rua localizados em área de terrenos baldios (código 4, importância 1) será pior para o usuário do que a rota que possui muitos segmentos de rua localizados em área industrial (código 2, importância 2).

Código	Descrição	O tempo é importante	Tipo do veículo
1	policial	<input checked="" type="checkbox"/>	1
2	policial	<input checked="" type="checkbox"/>	2
3	ambulancia	<input checked="" type="checkbox"/>	3
4	taxista	<input checked="" type="checkbox"/>	2
5	turista	<input type="checkbox"/>	2
6	cidadao	<input type="checkbox"/>	2

Figura 4.6: Perfis de usuários.

Código da categoria	Código da localização	Importância
4	4	1
5	1	3
5	2	2
5	3	4
5	4	1
6	1	4
6	2	3
6	3	2
6	4	1

Figura 4.7: Associações entre Perfis de Usuários e Características Descritivas dos Segmentos.

4.5.1.3 AS BIBLIOTECAS DE CASOS

O Módulo RBC possui, duas bibliotecas de casos: uma temporária e outra permanente³, sendo que os casos armazenados em cada uma não diferem em seus atributos mas nas suas formas de alimentação. O formato dos casos pode ser observado na figura Figura 4.8.

Analisando a Figura 4.8, observa-se que cada rota está associada a um conjunto de segmentos, a um ponto de origem e a um ponto de destino. A rota 1, por exemplo, está associada aos segmentos 88, 348, 349, 350 e 351 e, cada segmento está associado a um número que representa sua posição dentro da rota. Assim, nota-se que a rota 1

³ É importante ressaltar, que independentemente do sistema ser mono ou multi-usuário, existe apenas uma base de casos permanente, sendo que cada rota deve ser analisada cada vez que for recuperada, perante as características do perfil de usuário que a está requisitando.

possui a seguinte trajetória: 88, 351, 350, 349 e 348. O ponto de origem desta rota é o ponto 94 e o ponto de destino, o ponto 2.

A biblioteca de casos temporária, armazena as rotas que são geradas pelo Módulo Genético e que deverão ser analisadas pelo Módulo RBC. Cada vez que o Módulo Genético é chamado, esta base deve estar vazia para ser preenchida com as rotas geradas.

Rotas Permanentes - Tabela

Código	Tempo de execução	Hora de execução	Data de execução
2	9999	12:12	01/01/99
3	9999	12:12	01/01/99
4	9999	12:12	01/01/99
0	0		

Registro: 14 de 5

Rotas Segmento - Tabela

Código da rota	Código de segmento	Ordem na rota
1	88	1
1	348	5
1	349	4
1	350	3
1	351	2
2	15	2
2	29	4
2	30	5
2	31	6
2	32	7
2	33	8
2	34	9

Registro: 16 de 68

Rotas Permanentes Origem - Tabela

Código de rota	Origem da rota
1	94

Rotas Permanentes Destino - Tabela

Código de rota	Destino da rota
1	2

Figura 4.8: Formato dos casos da biblioteca⁴.

A biblioteca de casos permanente armazena as rotas que foram apresentadas ao usuário e por ele executadas. Inicialmente, quando o sistema opera pela primeira vez, esta base se encontra vazia, e é alimentada conforme o sistema vai sendo utilizado, caracterizando o aprendizado do sistema.

Algumas considerações devem ser feitas quanto aos atributos presentes nestas representações de casos.

Quanto à representação de rotas temporárias:

- o tempo de execução é uma estimativa feita sobre o comprimento real da rota e a velocidade máxima permitida, sem considerar questões de trafegabilidade, não sendo portanto, uma estimativa muito próxima da realidade;
- hora e data de execução da rota são consideradas a hora e data do sistema no momento de requisição da rota;

⁴ Os valores dos atributos: tempo, data e hora de execução são fictícios visto que, na realidade, as rotas não foram executadas. Elas foram inseridas na base permanente apenas para efeito de testes.

Quanto à representação de rotas permanentes:

- o tempo de execução deve ser instanciado pelo usuário após a execução real da rota;
- hora e data de execução são instanciadas também pelo usuário que executou a rota;

4.5.1.4 O CONJUNTO DE PROCEDIMENTOS

Os dados descritos nos itens anteriores são relacionados⁵ dentro do sistema com o objetivo de formular a rota. O usuário, ao requisitar uma rota, além de informar a origem e o destino requeridos, deve informar em qual perfil ele se encaixa. A partir desta informação, o sistema guiará todos os seus esforços para encontrar a rota, baseado nas características deste perfil.

Esse relacionamento feito entre os dados estão expressos na forma de funções e procedimentos dentro do código do sistema e geram informações importantes. Muitas destas gerações são feitas através de consultas em tempo de execução e não são armazenadas na base de dados. Outras são geradas e armazenadas podendo ser utilizadas em outras situações. Por exemplo, a ordem que cada segmento ocupa na rota está armazenada, já características tais como tipos de pavimentação e localização presentes nas rotas são reavaliadas sempre que a rota for executada.

Na avaliação das rotas, o número de segmentos com uma determinada característica, é considerado no quanto a rota avaliada se aproxima da rota ideal para o usuário. Portanto, é discutível até que ponto vale a pena reavaliar, em tempo de execução, cada uma das rotas.

Nas seções seguintes encontram-se mais detalhes sobre o conjunto de procedimentos, dentro de cada fase do processo de RBC, utilizados na resolução da problemática de planejamento de rotas.

⁵ Os relacionamentos existentes entre os dados podem ser melhor entendido através do Modelo Relacional apresentado no Apêndice A e dos algoritmos apresentados no Apêndice B.

4.5.1.5 INDEXAÇÃO

No sistema implementado, a indexação é feita pelos atributos “origem” e “destino” da rota. Contudo, outras características poderiam ser utilizadas para indexar os casos, no mesmo nível que estas ou em níveis diferentes.

O critério de indexação utilizado no PROTEUS, permite que se encontre a solução de um caso, dentro de uma solução (rota) existente na base permanente, que engloba a rota atualmente requerida. Por exemplo, considerando a rota 1 mostrada na Figura 4.8, percebe-se que esta engloba várias sub-rotas, entre elas estão as rotas compostas pelos seguintes segmentos: (88, 351, 350) ou (350, 349, 348). Estas últimas podem ser usadas como solução para alguma requisição feita ao sistema.

4.5.2 RECUPERAÇÃO

4.5.2.1 MÉTRICA DE SIMILARIDADE

Neste sistema, a métrica de similaridade utilizada, trabalha com a somatória de pesos atribuídos às características presentes nas rotas encontradas no processo de busca. Esta medida é aplicada a cada rota encontrada, em sua totalidade.

Para a aplicação da função que avaliará as rotas é considerado ainda, o perfil do usuário, sendo que para diferentes perfis, as características avaliadas pela equação pode mudar. A função de similaridade adotada é:

$$\text{Sim} = \left(1 - \left(\frac{1}{1 + \left(\frac{\partial}{\ell} \right)} \right) \right) * 100 \quad (4.1)$$

Em que:

- Sim = valor (em porcentagem) da similaridade do casos analisado ao problema de entrada;
- ∂ = valor inicial de similaridade entre a rota avaliada e a rota desejada;
- ℓ = comprimento total da rota;

$$\partial = \sum_{i=1}^n \frac{\sum_{j=1}^m \text{peso de } i \text{ em } j}{m} \quad (4.2)$$

Em que:

- ∂ = valor inicial de similaridade entre a rota avaliada e a rota desejada;
- i = característica a ser analisada;
- j = segmento a ser analisado;
- m = número de segmentos da rota avaliada;
- n = número de características a serem consideradas.

O valor de m varia de acordo com a rota avaliada, já que cada rota pode possuir um número de segmentos diferente e o valor de n varia de acordo com o perfil do usuário, pois para cada perfil, diferentes características são avaliadas.

Interpretando a métrica de similaridade em (4.1), observam-se alguns pontos:

- o denominador da divisão do segundo parêntese é sempre maior que 1 para que o resultado da divisão esteja no intervalo $[0,1]$;
- subtraímos o resultado da divisão do segundo parênteses de 1 para obter valores altos para casos muito similares;
- a multiplicação por 100 se deve ao fato que a similaridade deve ser dada em porcentagem.

Analisando a fórmula apresentada (4.2), percebe-se que a somatória mais interna sofre uma divisão pelo número de segmentos avaliados. Isto se faz necessário para que a medida de similaridade independa do tamanho da rota (em número de segmentos); o que ocasionaria dois problemas:

- rotas iguais (de acordo com suas características qualitativas) possuírem valores de similaridade diferentes, o que não é, a princípio, desejável⁶;
- rotas diferentes (de acordo com suas características qualitativas) possuírem valores de similaridade iguais.

Esta questão é melhor entendida observando a Figura 4.9. Atribuindo o valor 4 para a pavimentação em asfalto e para a localização comercial, para um determinado perfil, podemos fazer um cálculo simplificado da similaridade de cada uma das rotas:

A rota (b) é constituída pelos segmentos A, E, I, J e P (5 segmentos). Aplicando os devidos valores à fórmula (4.2), sem realizar a divisão pelo número de segmentos,

⁶ A princípio, não está sendo considerado que, um usuário possa preferir uma rota que passe por menos cruzamentos do que outra, porém, para levar esta restrição em conta, deveria-se estudar a retirada da divisão pelo número de segmentos da fórmula apresentada (4.2).

obtemos o valor 40, e seguindo o mesmo processo para a rota (c), composta pelos segmentos D, K, Q e R (4 segmentos) obtemos o valor 32.

Note que, a primeira rota, apesar de possuir as mesmas características da segunda rota, possui um valor de similaridade maior, sendo então, beneficiada pelo seu número de segmentos. Assim sendo, optou-se por utilizar o número de segmentos como um fator de normalização dos critérios de avaliação. Sendo assim, os valores deverão ser: 8 para a primeira rota e 8 para a segunda. A diferenciação das duas, virá na aplicação da fórmula (4.1), onde a distância real será levada em consideração como fator determinante da adequação da rota ao problema.

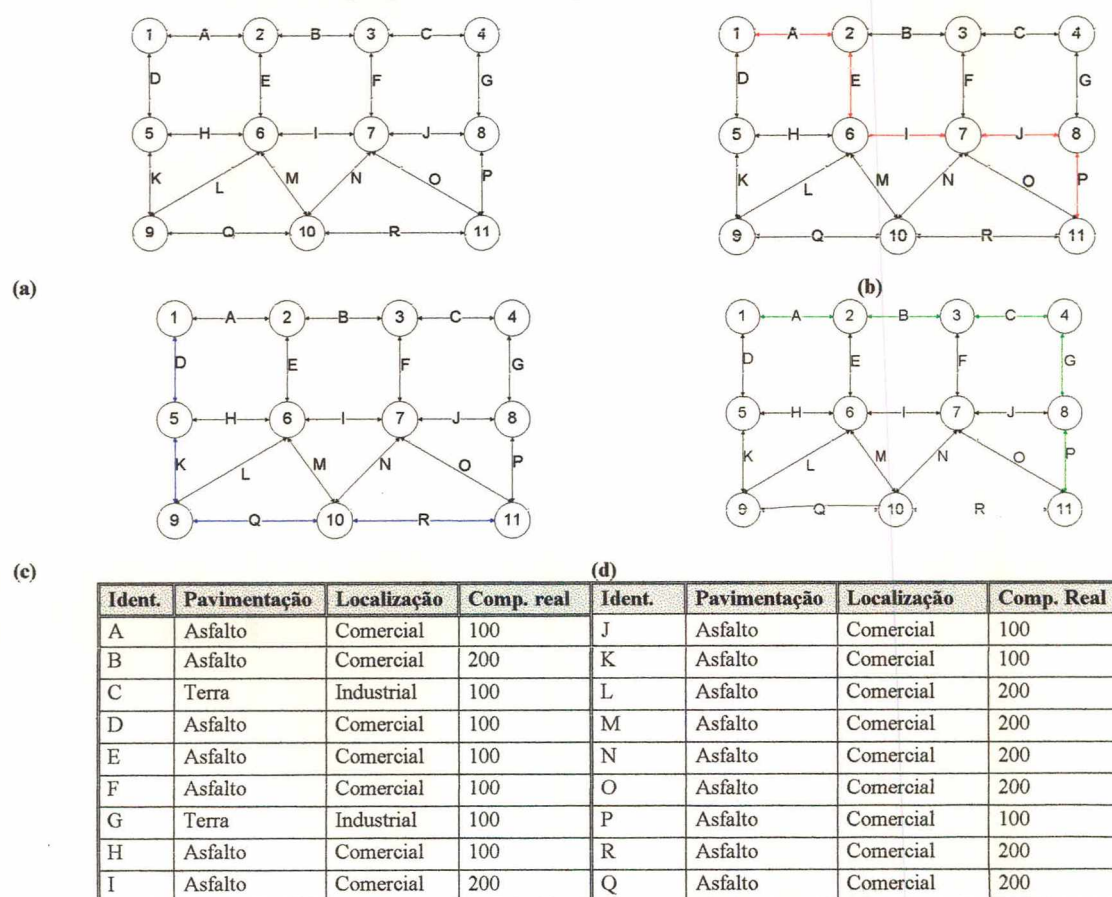


Figura 4.9: Exemplo para análise de rotas

Estudando agora, um segundo caso particular. Assuma que o valor 2 é atribuído para a pavimentação em terra e para a localização industrial e observe que a rota (d), formada pelos segmentos A, B, C, G e P (5 segmentos), será valorada com 32, o mesmo valor da rota (c). Analisando as características de cada uma, nota-se que a rota (d) possui um trecho (C,G) que deve ser penalizado, porém, devido ao número de segmentos deste trecho, esta rota tem o mesmo valor de similaridade da segunda rota, a

qual é, mais adequada. Ao dividir esse valores pelo número de segmento, consegue-se a diferenciação. A rota (c) obtém o valor 8 e a rota (d) obtém o valor 6,4.

Existe ainda uma terceira possibilidade onde rotas diferentes (de acordo com suas características qualitativas) e com número de segmentos diferentes, possuam o mesmo valor de similaridade mas, neste caso, a diferenciação é feita através da distância real. Se ainda assim, os valores de similaridades continuarem iguais, conclui-se que apesar das rotas serem diferentes, elas se adequam igualmente à solução procurada.

4.5.2.2 RECUPERAÇÃO

A recuperação das rotas para efeito de formar um conjunto de possíveis soluções é feita através de um limiar. Este limiar é um valor pré-estabelecido no sistema que representa o valor mínimo de similaridade que uma rota (caso) precisa apresentar, em relação à solução desejada (adequação às preferências/restrições do usuário) para ser recuperada. O sistema PROTEUS recupera no máximo três rotas, apresentando-as ao usuário, classificadas por ordem de adequação ao problema (valor de similaridade).

A recuperação, neste sistema, pode ser feita de duas formas:

- Recuperação completa: onde as rotas recuperadas têm a origem e o destino exatamente iguais ao problema de entrada;
- Recuperação de trecho de rota: onde as rotas recuperadas contêm a rota desejada, isto é, a origem e o destino do problema de entrada podem ser obtidos dentro das rotas recuperadas.

4.5.2.3 BEST MATCH

O *Best Match* é realizado de forma simples, através da escolha dos três casos mais similares (com valores mais altos de similaridade). É interessante dizer que, como o sistema gera/recupera várias rotas, é possível que rotas de mesma similaridade sejam geradas/recuperadas. Neste caso, optou-se por escolher uma das rotas dentre as várias que possuam o mesmo valor de similaridade.

4.5.3 ADAPTAÇÃO

A adaptação, neste sistema, é requerida sempre que as rotas geradas ou recuperadas possuem trecho que apresentam características indesejáveis para o usuário. Neste sentido, o sistema tenta modificar a rota, substituindo este trecho indesejável por

outro, que possua características mais adaptadas às preferências do usuário.

Para realizar a adaptação optou-se por primeiramente avaliar a relação custo/benefício. A idéia implementada no sistema é composta dos seguintes passos:

- na análise da rota para estabelecimento da similaridade, algumas observações a mais são realizadas. Em cada segmento, cada característica analisada que não satisfizer as preferências do usuário, resulta em uma marca que servirá para descobrir os trechos das rotas que são indesejáveis;
- após a verificação da existência de trechos compostos de um número fixo (pré-estabelecido no sistema) de segmentos adjacentes que não são satisfatórios, a rota passa a ser candidata a ser adaptada;
- usuário então, decide se deseja que o sistema tente adaptar este trecho. Se for desejável, o processo de encontrar a rota é novamente iniciado, passando como entrada o ponto de origem e o ponto de destino deste trecho;
- em seguida, a rota a ser adaptada deve ser montada, substituindo o trecho ruim, pela rota então recuperada e uma nova avaliação da rota inteira é realizada;
- apenas, uma nova chance de se encontrar uma rota mais satisfatória deve se concretizar para cada uma das três rotas apresentadas ao usuário.

Esta forma de adaptação pode ser classificada como substituição baseada em casos, já que utiliza-se casos para valorar atributos que não satisfazem algumas restrições.

4.5.4 AVALIAÇÃO E APRENDIZADO

No sistema PROTEUS, a avaliação da solução encontrada é realizada pelo próprio usuário, desejavelmente, após a execução real da rota. Sendo assim, a opção por realizar reparos de forma automatizada (modificação da rota através de procedimentos implementados no sistema), mesmo que, a partir de especificações do usuário, não foi considerada, visto que, o que geralmente pode fazer da rota encontrada, uma solução inadequada, seriam fatores externos e imprevisíveis a modelagem do sistema tais como: obras na malha viária, acidentes etc.

O aprendizado, se dá através da incorporação de rotas geradas pelo módulo genético ou formuladas a partir do processo de adaptação, sempre com o aval do usuário. É ele quem decide se a rota deve ou não ser incorporada à base de casos de

acordo com a sua satisfação após a execução real da mesma.

5.2 O MÓDULO GENÉTICO

Para ilustrar a performance do Módulo Genético na geração de rotas, executou-se o sistema com as seguintes configurações para as variáveis do Algoritmo Genético (AG):

- Número de gerações = 30 : indicando que as rotas deverão ser geradas dentro de, no máximo, 30 iterações do AG;
- Número de cromossomos = 10 : indicando que o AG tentará encontrar 10 rotas para uma determinada requisição do usuário;
- Origem = 1 : indicando que a rota começará no cruzamento entre as ruas Juscelino Kubstchek e Natal (Figura 5.2).
- Destino = 335 : indicando que a rota terminará no cruzamento entre as ruas Santa Catarina e Salgado Filho (Figura 5.2).
- Perfil do usuário = Taxista (código 4) : indicando o perfil considerado, para que o AG possa escolher sobre qual grafo agir (orientado ou não-orientado).

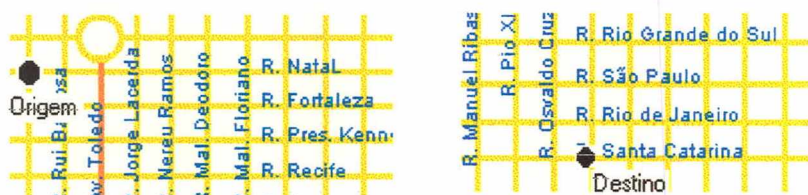


Figura 5.2: Pontos de origem e destino da rota de teste do AG.

É importante salientar que o perfil do usuário é uma variável com uma relevância maior para o Módulo RBC do que para o Módulo Genético, visto que a avaliação de rotas é diretamente dependente do perfil, e a geração pode ocorrer sem a especificação deste. Porém, como existem perfis para os quais as ruas de mão única são desconsideradas (bombeiros, policiais, ambulâncias), é desejável que o algoritmo considere, nestes casos, um grafo não-orientado para aumentar as possibilidades de rotas diferentes.

A título de demonstração, a realização de testes para este perfis não mostrariam a eficiência do algoritmo no que diz respeito à obediência à restrição de ruas mão única, por isso o perfil de taxista foi escolhido para esta exemplificação.

Analisando a Figura 5.3, nota-se na parte inferior, uma barra de progressão, que indica o número de iterações que o algoritmo executou para gerar as 10 rotas. No caso

desta execução, o algoritmo levou 17 iterações para conseguir concluí-las. Apesar desta figura só mostrar uma rota (curva em azul), as 09 rotas restantes já estão geradas e a plotagem destas, se dá, uma a uma e após a geração de todas elas.

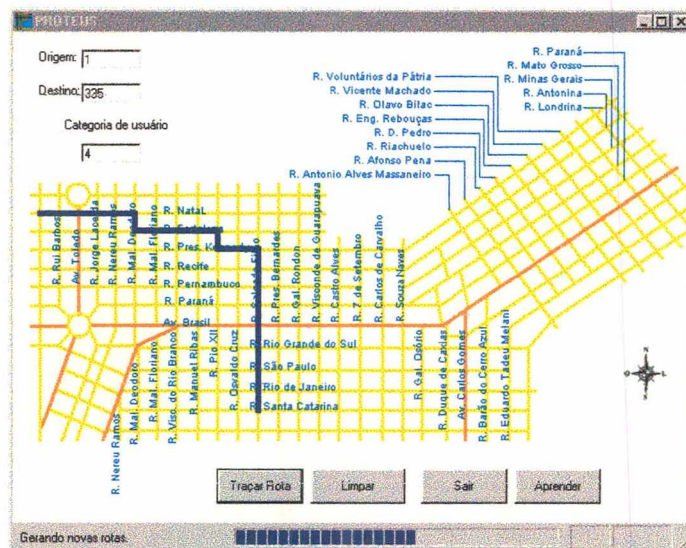


Figura 5.3: Primeira rota gerada

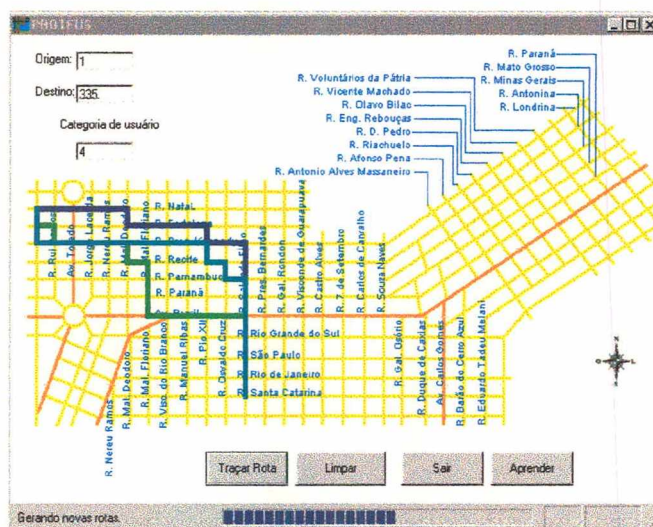


Figura 5.4: Três rotas geradas

A Figura 5.4 mostra as três primeiras rotas geradas (curvas: azul, ciano e verde) onde podemos notar que, apesar de alguns segmentos se sobreporem, elas apresentam

vários trechos distintos, o que ocasionará uma diferença nas suas avaliações, de acordo com as características destes trechos.

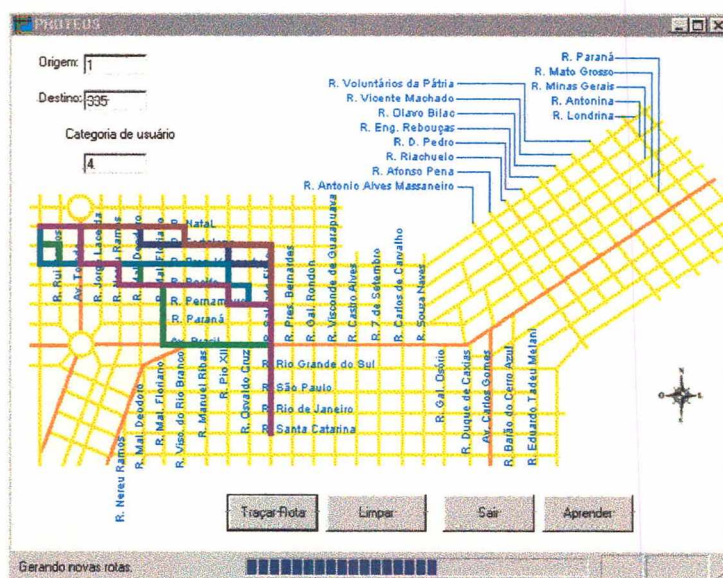


Figura 5.5: Cinco rotas geradas

Cinco diferentes rotas podem ser observadas na Figura 5.5 (curvas: azul, ciano, verde, vermelho e lilás). Na Figura 5.6, observa-se 09 diferentes rotas (curvas numeradas) sendo que todas elas se sobrepõem nos quatro últimos segmentos. Isto não significa que não existem outros caminhos para chegar ao destino, que não passem por estes quatro últimos segmentos. A causa desta sobreposição é atribuída à aleatoriedade do AG, já que em outras execuções do sistema, para as mesmas instâncias das variáveis, rotas que não passam por estes segmentos foram geradas, como mostra na Figura 5.7, as rotas representadas nas cores azul e verde.

Note ainda que uma das cinco últimas rotas geradas sobrepõe totalmente a rota representada, na Figura 5.5, pela curva vermelha, já que não é mais possível enxergá-la na Figura 5.6.

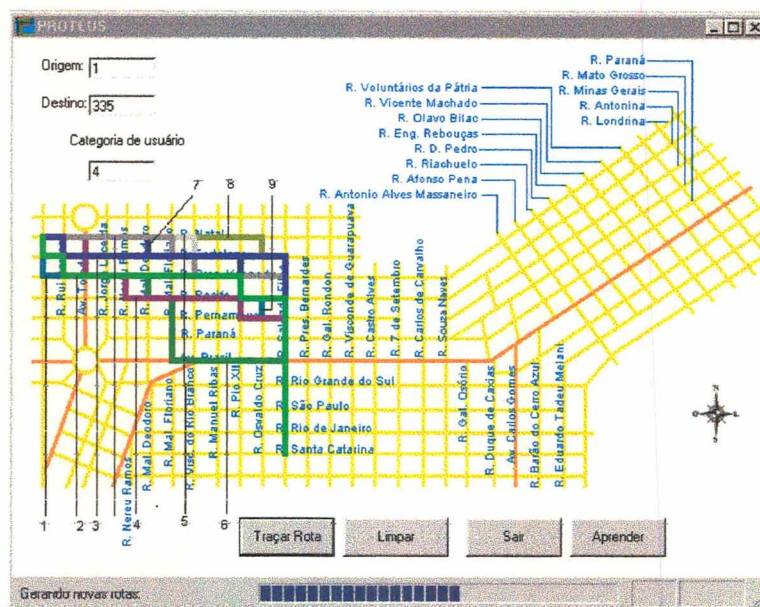


Figura 5.6: Todas as rotas geradas.

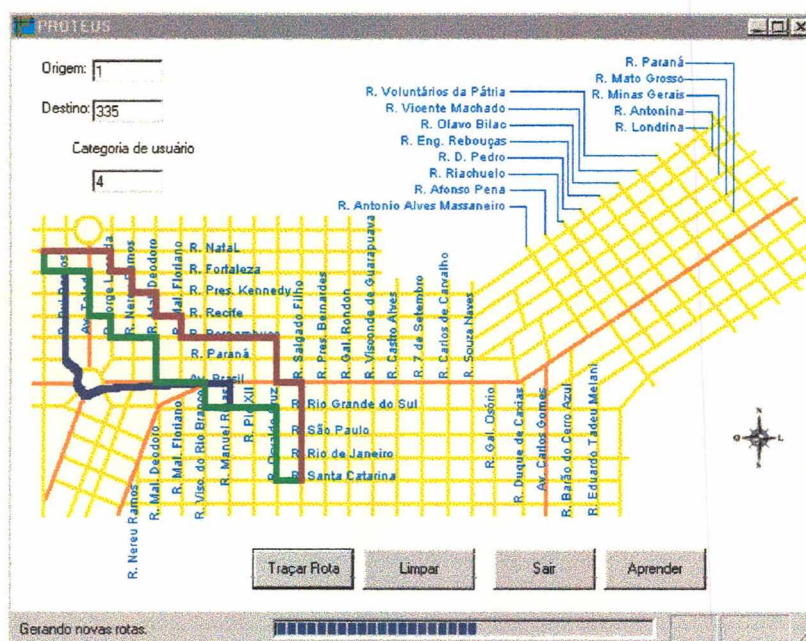


Figura 5.7: Rotas com o trecho final diferente.

5.3 O MÓDULO DE RACIOCÍNIO BASEADO EM CASOS

Para ilustrar a performance da aplicação de Raciocínio Baseado em Casos na avaliação e recuperação de rotas, diferentes exemplos são mostrados para abordar

diferentes situações, diferentes fluxos de controles do algoritmo e diferentes resultados.

A Figura 5.8 ilustra uma situação onde o sistema tentou recuperar uma rota completa a partir das entradas do usuário. Neste momento nenhuma rota armazenada na base de casos, tem seus atributos de origem e destino instanciado com valores iguais às entradas fornecidas. A Figura 5.9 ilustra a tentativa de recuperar rotas que contenham os pontos de origem e destino fornecidos pelo usuário, ou seja, o sistema está procurando uma rota que contenha a rota procurada (recuperação parcial). Desta forma, apenas o trecho correspondente à rota desejada será recuperado e analisado. Também neste caso, nenhuma rota foi recuperada da base de rotas.

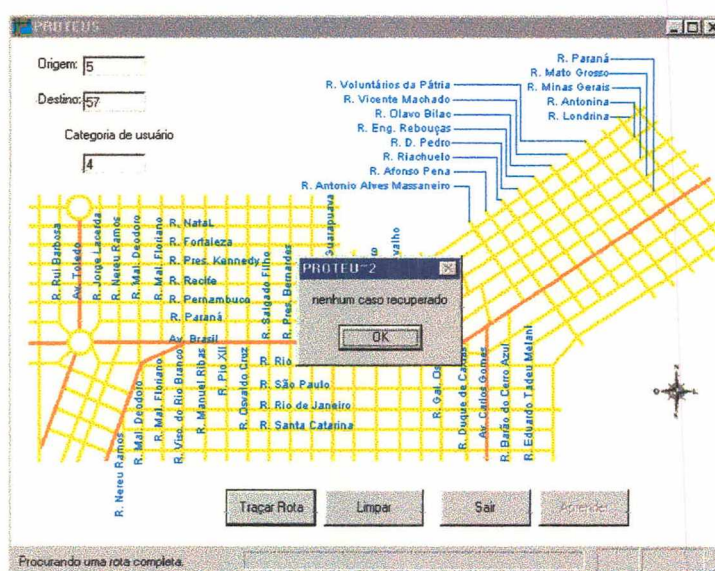


Figura 5.8: Tentativa de recuperação completa.

Sobre as rotas geradas pelo Módulo Genético, é necessário avaliar cada uma delas quanto as suas características de acordo com as preferências do usuário. Executando o módulo AS (5 cromossomos), foram obtidas 5 rotas, sendo que as três melhores tiveram seus valores de similaridade de acordo com a Figura 5.10.

O fato do sistema ter escolhido duas rotas de mesmo valor de similaridade se deve ao fato que, das 5 rotas obtidas, algumas possuíam o mesmo valor de similaridade. O sistema opta por escolher rotas com valores diferentes, porém, se apenas dois valores são obtidos para todas as rotas, o sistema escolhe as de maior valor de similaridade.

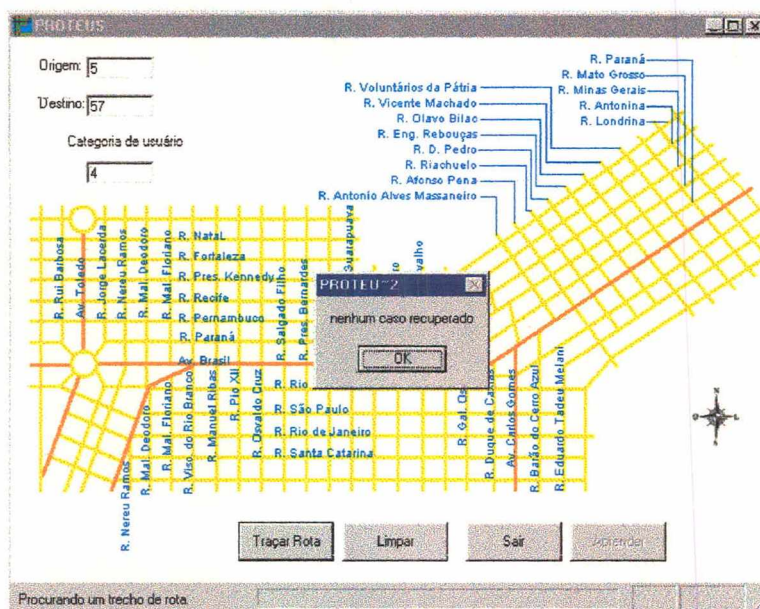


Figura 5.9: Tentativa de recuperar a rota como um trecho de outra.

Note que as medidas de similaridade da rota 4 e da rota 5 são iguais, apesar de serem rotas fisicamente distintas como mostrado na Figura 5.11, onde a rota 2 corresponde à curva azul, a rota 4 corresponde à curva vermelha e a rota 5 corresponde à curva verde. Essa igualdade de similaridades se deve a uniformidade encontrada nas características dos segmentos da malha viária.



Figura 5.10: Similaridade das três rotas escolhidas.

No exemplo seguinte, observa-se o comportamento do sistema quando rotas que já foram geradas antes e se encontram armazenadas na base de casos, atendem à uma requisição do usuário. Neste caso o sistema recuperou três rotas (as três melhores rotas que foram armazenadas a partir da geração mostrada na Figura 5.11). A Figura 5.12 e a Figura 5.13 ilustram a resposta do sistema quanto à primeira rota recuperada e sua similaridade. Depois de mostrado todas as rotas e suas similaridades, o sistema plota as rotas para o usuário.

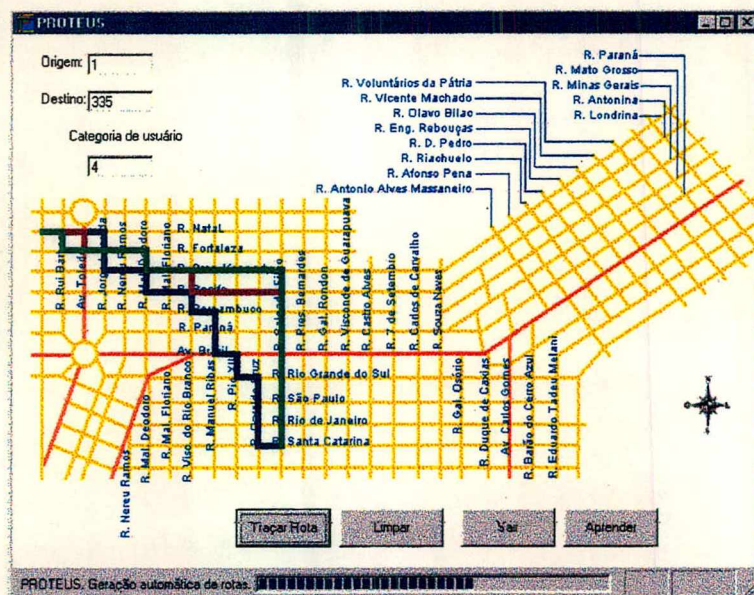


Figura 5.11: Rotas escolhidas.

É importante acrescentar que o sistema recupera todas as rotas e calcula suas similaridades de acordo com o problema atual mas, plota apenas as três melhores. As mensagens mostradas, por exemplo, nas figuras Figura 5.12 e Figura 5.13 são apenas para ilustrar o funcionamento do sistema e devem ser retiradas para otimizar a interface do sistema para com o usuário.

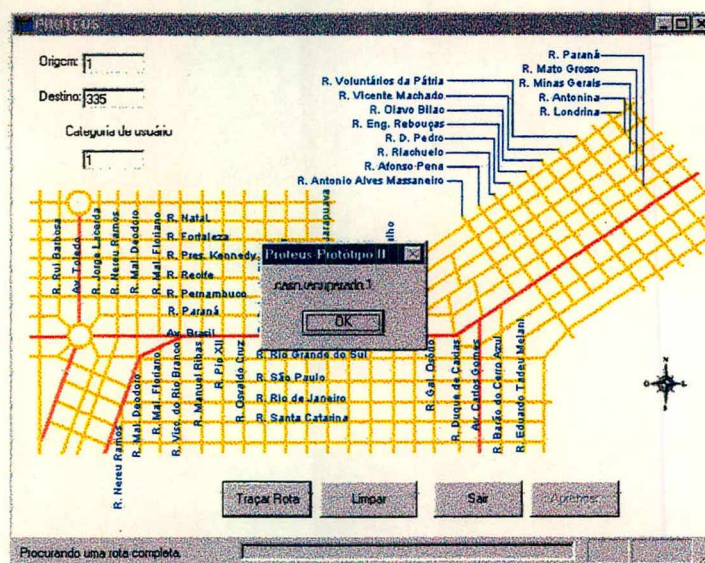


Figura 5.12: Primeiro caso recuperado.

Figura 5.13: Valor de Similaridade do primeiro caso recuperado.

É importante notar que, as similaridades são diferentes quando se trata de perfis de usuários diferentes (taxista e policial motociclista). A Figura 5.13 ilustra o resultado de uma execução do sistema onde, foram mantidos os valores de “origem” e “destino” e modificado o perfil de usuário. A rota recuperada neste exemplo, onde o perfil escolhido é “policial motociclista” (código 1) é a rota mostrada pela Figura 5.11 (rota azul). Na execução ilustrada pela Figura 5.11 e Figura 5.10, o perfil escolhido é outro (taxista) e o valor de similaridade, é diferente do valor de similaridade apresentado pela execução ilustrada na Figura 5.13.

A Figura 5.14 mostra, outro exemplo de execução do sistema onde o resultado foi uma recuperação parcial. A rota requerida não foi encontrada na base de casos e, o sistema passou a procurá-la dentro das rotas existentes. Três rotas foram recuperadas e uma delas é mostrado pela Figura 5.15.

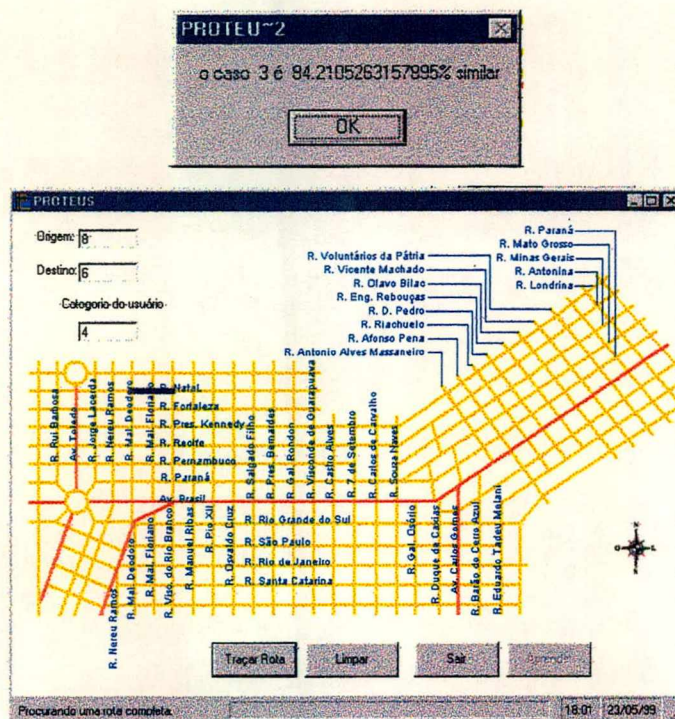


Figura 5.15: Exemplo de rota e sua similaridade.

Segmentos: Tabela							
Código	Origem	Destino	Mão Única?	Pavimentação	Localização	Velocidade	Comprimento
7	7	8	<input type="checkbox"/>		1	1	1
8	8	9	<input type="checkbox"/>		1	1	1
Registro: 14 de 649							

Segmentos: Tabela				
Preferência na Origem?	Preferência no Destino?	Semáforo na Origem?	Semáforo no Destino?	Lombada
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Registro: 14 de 649				

Figura 5.16: Características dos segmentos que compõe a rota.

Na Figura 5.16, observa-se que as características dos segmentos estão instanciada através de códigos ou campos lógicos.

Quanto ao aprendizado, o sistema considera a opinião do usuário. A Figura 5.17 ilustra o ambiente onde o usuário pode informar ao sistema sua intenção de acrescentar rotas a base de casos. O usuário pode escolher quais das três melhores rotas será aprendida.

A opção por escolher duas rotas com mesmo valor de similaridade se deve, principalmente, ao fato que apesar das duas rotas possuírem o mesmo valor elas são compostas por segmentos diferentes e podem ter valores de similaridade diferentes quando se trata de outros perfis.

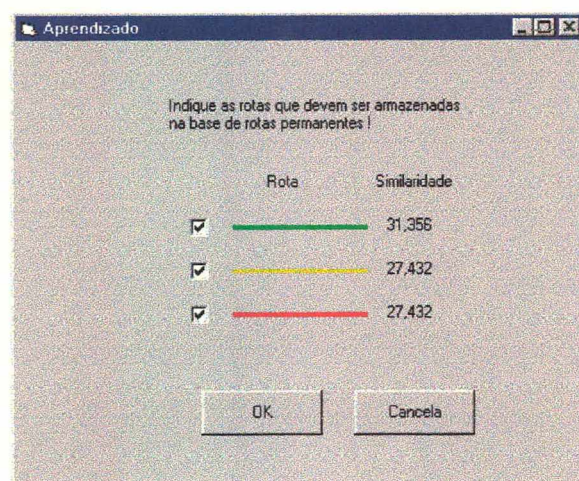


Figura 5.17: Aprendizado do sistema.

6 CONCLUSÃO

6.1 CONSIDERAÇÕES FINAIS

Neste trabalho, foi implementado um sistema híbrido para gerar rotas válidas que atendam as características de seus usuários.

A vantagem da utilização de um modelo híbrido está na questão de otimização no processo de resolução do problema. No sistema desenvolvido, a técnica de Algoritmos Genéticos é utilizada somente quando a geração de novas rotas se fizerem necessária.

A técnicas de Algoritmos Genéticos mostrou-se vantajosa uma vez que ela fornece diversas soluções (rotas) para serem analisadas, mas nada impede que se use técnicas diferentes, inclusive técnicas que não são da Inteligência Artificial, tais como Pesquisa Operacional, Teoria dos Grafos etc, com apresentado no Capítulo 3.

O uso de RBC para análise das rotas se mostra eficiente, principalmente no que tange à fidelidade ao processo de raciocínio humano para este tipo de problema. As soluções apresentadas, se mostraram adequadas ao proposto inicialmente. No Capítulo 5, foram feitas simulações que mostraram a adequação das respostas aos problemas apresentados.

O tempo gasto para recuperação de soluções é um tempo aceitável mediante a base de casos (cerca de 40 rotas) que atualmente se tem no sistema. O crescimento da base pode acarretar em uma demanda de tempo maior para obtenção da solução, porém, isto aconteceria para qualquer outra técnica que tentasse a recuperação visto que, trata-se aí de uma questão de performance do mecanismo de recuperação da informação do sistema gerenciador de banco de dados utilizado.

O tempo gasto na análise da similaridade das rotas pode ser otimizado. Alterações no código que implementa o cálculo da similaridade ou mesmo na própria função de similaridade podem diminuir o tempo de resposta do sistema.

Grande parte do tempo gasto no desenvolvimento deste trabalho se reverteu à modelagem do problema de planejamento de rotas dentro das técnicas propostas.

Teoricamente e filosoficamente, as técnicas se mostraram muito suscetíveis ao sucesso para a resolução deste problema, porém, como em qualquer outro domínio, um estudo específico sobre questões de transporte e mais objetivamente, sobre planejamento de rotas, foi necessário para que se atendessem realmente às restrições clássicas do problema através de uma modelagem adequada.

Dentro das vantagens oferecidas pela técnica de RBC deve-se também lembrar da questão de justificativa. Um usuário que obtém uma rota recuperada a partir de uma base que vem sendo alimentada com a aprovação dos próprios usuários, tende a confiar mais na adequação da solução encontrada. Além disso, pode-se construir um módulo de justificativa que apresenta ao usuário, argumentos que fundamente a escolha de determinadas rotas.

Uma outra vantagem deste sistema é que ele faz uma análise de rotas a nível de segmentos de rua, o que torna a avaliação mais precisa e próxima da realidade. Esta questão foi brevemente discutida em (LIU, 1996), onde o autor comenta a desejabilidade de tal funcionalidade.

Outro aspecto importante que deve ser ressaltado, é que uma mudança física na malha viária pode invalidar várias rotas que estão armazenadas na base de rota, portanto, a cada mudança realizada nas matrizes que representam o espaço geográfico, as rotas na base de casos devem ser revistas para garantir a sua real possibilidade de execução.

O sistema PROTEUS foi inicialmente proposto para atuar sobre malhas viárias urbanas mas pode ser estendido para outros espaços similares, bastando para isto, modelar o espaço em um grafo e definir as características que devem ser avaliadas.

A extensibilidade do sistema se pronuncia nas seguintes possíveis aplicações:

- No âmbito rodoviário: empresas de ônibus, empresas de entrega intra cidades, viajantes em geral;
- No âmbito ferroviário: empresas de transporte ferroviário;
- No âmbito marítimo e no âmbito aéreo: organizações responsáveis pela navegação marítima e aérea.

Dentro do domínio urbano foram definidas algumas classes de usuários que caracterizam algumas aplicações: motoristas de taxis, turistas, entregadores que utilizam motos, entregadores que utilizam carro, motoristas de caminhão, ambulâncias,

bombeiros, policiais, motoristas comuns. O que diferencia estas aplicações são apenas as características consideradas relevantes para análise das rotas, sendo que estas categorias estão definidas dentro do sistema PROTEUS como perfis.

6.2 TRABALHOS FUTUROS

No sentido de melhorar alguns pontos no sistema PROTEUS, sugere-se:

- Tradução da implementação atual para a linguagem portátil, para executar em diversos tipos de arquitetura, onde testes de performance poderão ser realizados. Esta tradução deverá ser acompanhada de uma otimização de código, com o intuito de diminuir a complexidade deste;
- Inclusão de mais informações dentro da malha viária tais como: localização de shoppings, restaurantes, escolas e hospitais de forma a fornecer ao usuário, uma maior facilidade de localização dentro da cidade ampliando o escopo de utilização do sistema ou ainda a inclusão de mais características para serem analisadas tais como: índices de acidentes, periculosidade da área de localização etc.
- Incorporação de um sistema baseado em Lógica Difusa (KLIR, 1995), para modelar características de tempo, hora e sazonalidade. A idéia inicial deste sistema envolve um estudo mais detalhado sobre a modelagem destas características, inclusive no que tange a diferença de opiniões que existem no domínio desta aplicação. Um sistema que consiga incorporar a opinião de vários especialistas, retirando desta combinação uma ponderação sobre um comportamento aproximará, ainda mais, o processo inerente a este sistema de um processo de raciocínio humano;
- Implementação de um sistema baseado em Agentes Inteligentes (WOOLDRIGE, 1995), que possam representar o próprio usuário do sistema, ajudando desta forma na realização de testes para incorporação do sistema em diferentes espaço de problema, isto é, todas as vezes que for instalado sobre uma malha viária diferente, possa-se testá-lo e incrementar sua base de casos com algumas situações iniciais. Esses agentes deverão ser dotados de uma capacidade de lidar com raciocínio de senso comum (DURKIN, 1994), para garantir uma boa aproximação da realidade, inclusive no que tange à real possibilidade de gerar uma mudança na rota em tempo de execução;
- Implementação envolvendo execuções paralelas do sistema, onde uma

instância do sistema seja capaz de conhecer o que outra instância está planejando, com o intuito de apresentar soluções que considerem a dinamicidade do tráfego dentro de uma malha viária urbana.

- Adaptação do sistema desenvolvido para resolução de problemas similares tais como: roteamento de informações em redes de computadores, distribuição de energia em redes elétricas entre outras.

REFERÊNCIAS BIBLIOGRÁFICAS

- AHO, A. V.; ULLMAN, J. D.. **Foundations of Computer Science**. C. ed. New York: Computer Science Press, 1995. p. 786.
- ASSAD, A. A., Modeling and Implementation Issues in Vehicle Routing. In: GOLDEN, B. L.; ASSAD, A. A.. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Cience Publishers B.V,1998. p. 7-45.
- BALL, M. O., Allocation/Routing: Models and Algorithms. In: GOLDEN, B. L.; ASSAD, A. A.. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Cience Publishers B.V,1998. p. 199-221.
- BERSTEIN, D.; KORNHAUSER, A., Map Matching for Personal Navigation Assistants. In: TRANSPORTATION REASEARCH BOARD, 77TH ANNUAL MEETING. Washington:1998.
- BURMEISTER, B.; HADDADI, A.; MATYLIS, G. Application of multi-agent systems in traffic and transportation. **IEE Proccedings - Software Enginnering**. v. 144, n. 1, p. 51-60. feb. 1997.
- CARVALHO, A. P. L. F.; BRAGA, A. P.; LUDERMIR, T. B. **Fundamentos de Redes Neurais Artificiais**. 1^a. ed. Rio de Janeiro: 11^a Escola de Computação, Imprinta Gráfica e Editora Ltda, 1998. p. 246.
- CORMEN, T. H.; LEISERSON, C. R.; RIVEST, R. L. **Introduction to Algorithms**. 1^a. ed. USA:The MIT Press, 1990. p.1028.
- CASTRO, J. P. **Um algoritmo Evolucionário para Geração de Planos de Rotas**. Florioanópolis, 1999. Dissertação apresentada para obtenção do grau de Mestre em Engenharia de Produção e Sistemas, Universidade Estadual de São Paulo. p.81.
- CHENG Q., WEI H.Y.; **Designing Pharmaceuticals Integrating Case-Based Reasoning and Decision Theory**. Costas Tsatsoulis – Unversitu of Kansas.

- CHRISTOFIDES, N. Vehicle routing. In: LAWLER, L. E. *et al.* **The Traveling Salesman Problem: a guided tour of combinatorial optimization**. Great Britain: John Wiley & Sons, 1985. p. 431- 448.
- DONALD, M.. *Precis of Origins of the Modern Mind: Three Stages in the Evolution of Culture and Cognition. Behavioral and Brains Science.* n. 16. v. [4]: p.737 – 791. 1997.
- DURBIN, R.; WILLSHAW, D.. An analogue approach to the travelling salesman problem using na elastic net method. *Nature*, v. 626, p. 689-691, apr. 1987.
- DURKIN, J.. **Expert Systems: Design and Development**. 1^a. ed. New Jersey: Prentice Hall, Inc, 1994. p. 800.
- GÄRLING, T., Behavioral Assumptions Overlooked in Travel-Choice Modelling. In: SEVENTH INTERNATIONAL CONFERENCE ON TRAVEL BEHAVIOUR (IATBR-1994). Santiago: jun. 1994. p. 115-126.
- GARFINKEL, R., S.. Motivation and modeling. **The Traveling Salesman Problem: a guided tour of combinatorial optimization**. Great Britain: John Wiley & Sons, 1985. p. 1- 15.
- GOLDEN, B. L.; ASSAD, A. A.. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Science Publisher B.V., 1988a. p. 479.
- GOLDEN, B. L. , Route Planning for Coast Ships. In: GOLDEN, B. L.; ASSAD, A. A.. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Cience Publishers B.V,1998b. p 439-443.
- GOONATILAKE, S.; KHEBBAL, S.. **Intelligent Hybrid Systems**. 1^a. ed. England: John Wiley & Sons, 1995. p. 325.
- HAIGH, K. R.; VELOSO, M. M.. Combining Search and Analogical Reasoning in Path Planning from Road Maps. In AAAI-94 WORKSHOP. **Case-Based Reasoning: Working Notes**. Washington: AAAI Press, jul. 1993. p. 182-187.
- HAIGH, K. R.; SHEWCHUCK, J. R.; VELOSO, M. M.. Exploiting Domain Geometry in Analogical Route Planning. **Journal of Experimental and Theoretical Artificial Intelligence**, n. 9, p. 509-541, 1997.

- HAIGH, K. R.; SHEWCHUCK, J. R.; VELOSO, M. M. Route Planning and Learning from Execution. In: AAAI FALL SUMPOSIUM. **Planning and Learning: On to Real Applications: Working Notes**. New Orleans: AAAI Press., nov. 1994. p. 58-64.
- HAIGH, K. R.; VELOSO, M. M. Route Planning by Analogy. In: ICCBR-95. **Proceedings Case-Based Reasoning Research and Development**. Sisembra: Springer-Verlag, oct. 1995. p. 169-180.
- HILLIER, S. H; GERALD, J. L.. **Introduction to Operations Research**. 1^a. ed. USA: McGraw-Hill, 1995. p. 986.
- HOFFMAN, A. J.; WOLFE, P., History. In: LAWLER, L. E. *et al.* **The Traveling Salesman Problem: a guided tour of combinatorial optimization**. Great Britain: John Wiley & Sons, 1985. p. 1- 15.
- INFOTEL www.infotel.it/atlantic-sharks/planner.htm
- JAILLET, P., The Probabilistic Vehicle Routing Problem. In: GOLDEN, B. L.; ASSAD, A. A.. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Cience Publishers B.V,1998. p 293-318.
- JOHNSON, D. S.; PAPADIMITRIOU, C. H, Computational complexity. In: LAWLER, L. E. *et al.* **The Traveling Salesman Problem: a guided tour of combinatorial optimization**. Great Britain: John Wiley & Sons, 1985a. p. 37-85.
- JOHNSON, D. S.; PAPADIMITRIOU, C. H., Performance guarantees for heuristics. In: LAWLER, L. E. *et al.* **The Traveling Salesman Problem: a guided tour of combinatorial optimization**. Great Britain: John Wiley & Sons, 1985. p.145-179.
- KLIR, G. J.; YUAN, B.. **Fuzzy Sets And Logic: Theory and Applications**. 1^a. ed. New Jersey: Prentice Hall PTR, 1995. p. 574.
- KOLODNER, J.. **Case-based reasoning**. 1^a. ed. Harcover: Morgan Kaufman Publisher, 1993. p. 612.

- KOZA, J. R.. **Genetic programming: on the programming of computers by means of natural selection**. 1^o. ed. USA: The MIT Press, 1992. p. 819.
- LAWLER, L. E. *et al.* **The Traveling Salesman Problem: a guided tour of combinatorial optimization**. Great Britain: John Wiley & Sons, 1985. p. 465.
- LEE, O.. **Passeios e conexidade em grafos mistos: algoritmos e complexidade computacional**. São Paulo, 1994. Dissertação apresentada para obtenção do grau de Mestre em Matemática Aplicada - Instituto de Matemática e Estatísticas – Universidade Estadual de São Paulo. p.118.
- LIU, B.. **Using Knowledge to Isolate Search in Route Finding**. 1995
- LIU, B.. **Intelligent Route Finding: Combining Knowledge, Cases and an Efficient Search Algorithm**. In: 12TH EUROPEAN CONFERENCE ON ARTIFICIAL INTELLIGENCE. Local: John Wiley & Sons Ltd., 1996.
- MEDSKER, L. R.. **Hybrid Intelligent Systems**. 1^o. ed. USA: Kluwer Academic Publishers, 1995. p. 289.
- MÜHLENBEIN, H.. **Evolution in Time and Space – The Parallel Genetic Algorithm**. In Autor. **Foundation of Genetic Algorithms**. ** ed. USA: Morgan Kaufmann Publishers, Inc, 1991.
- NAKAMITI, G.; GOMIDE, F.. **An evolutive fuzzy mechanism based on past experiences**. In: SECOND EUROPEAN CONGRESS ON INTELLIGENT TECHNIQUES IN SOFT COMPUTING. Aachen, sep, 1994. p. 1211-1217.
- PANDYA, A., S.; MACY, R., B.. **Pattern Recognition with Neural Networks in C++**. 1^o. ed. USA: CRC Press, 1996. p. 410.
- PERES, S.M. *et al.* **A Hybrid System For Route Planning**. In AAAI Spring Symposium Series. Technical Report, Stanford, 1997. p. 112-116.
- POWELL, W. B., **A Comparative Review of Alternative Algorithms for the Dynamic Vehicle Allocation Problems**. In: GOLDEN, B. L.; ASSAD, A. A.. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Science Publishers B.V., 1998. p. 249-291.

- PRESSMAN, R. S.. **Engenharia de Software**. 3^a. ed. São Paulo: Makron Books, 1995. p. 1056.
- PSARAFITS, H. N., Dynamic Vehicle Routing Problems. In: GOLDEN, B. L.; ASSAD, A. A. **Vehicle Routing: Methods and Studies**. Amsterdam: Elsevier Cience Publishers B.V,1998. p. 223-248.
- RABUSKE, M. A.. **Introdução à Teoria dos Grafos**. 1^a ed. Florianópolis, Editora da UFSC, 1992. p. 173.
- RICH, E.; KNIGHT, K.. **Inteligência Artificial**. 2^a. ed. São Paulo: Makron Books, 1993. p. 722.
- RICHTER, M. M.. **Introduction**. In LENZ, M. *et al.* **Case-Based Reasoning Technology: from Foundations to Applications**. 1^a. ed. Germany: Springer-Verlag, 1998. p. 01-15.
- RUSSEL, S. J.; NORVIG, P.. **Artificial Intelligence: A Modern Approach**. 1^a. ed. USA: Prentice Hall, 1995. p. 932.
- SCOTT, K.; JIMÉNEZ, G. P.; BERSTEIN, D.. Finding Alternatives to the Best Path. In TRANSPORTATION RESEARCH BOARD, 76TH ANNUAL MEETING. Washington: 1997a.
- SCOTT, K.; BERSTEIN, D.. Solving a Best Path Problem when the Value of Time Function is Nonlinear. In TRANSPORTATION RESEARCH BOARD, 76TH ANNUAL MEETING. Washington: 1997a..
- SHEN Z.; LUI H. C.; DING L.. **Approximate Case-Based Reasoning on Neural Networks**. Singapore: National University of Singapore.
- STATHOPOULOS, A.; PAPOUTSI, E., Understanding Drivers Route Choice Behaviour: An Experimental Approach.. In: SEVENTH INTERNATIONAL CONFERENCE ON TRAVEL BEHAVIOUR (IATBR-1994). Santiago: jun. 1994. p. 151-163.
- SUN, R.. Robust reasoning: integrating rule-based and similarity-based reasonig. In *Artificial Inteligence*, Alabama, n. 75, p. 241-295, 1995.

WEBER, R.. **CBR Course Home Page**. Notas de Aula. 1997
<http://www.eps.ufsc.br/~martins>

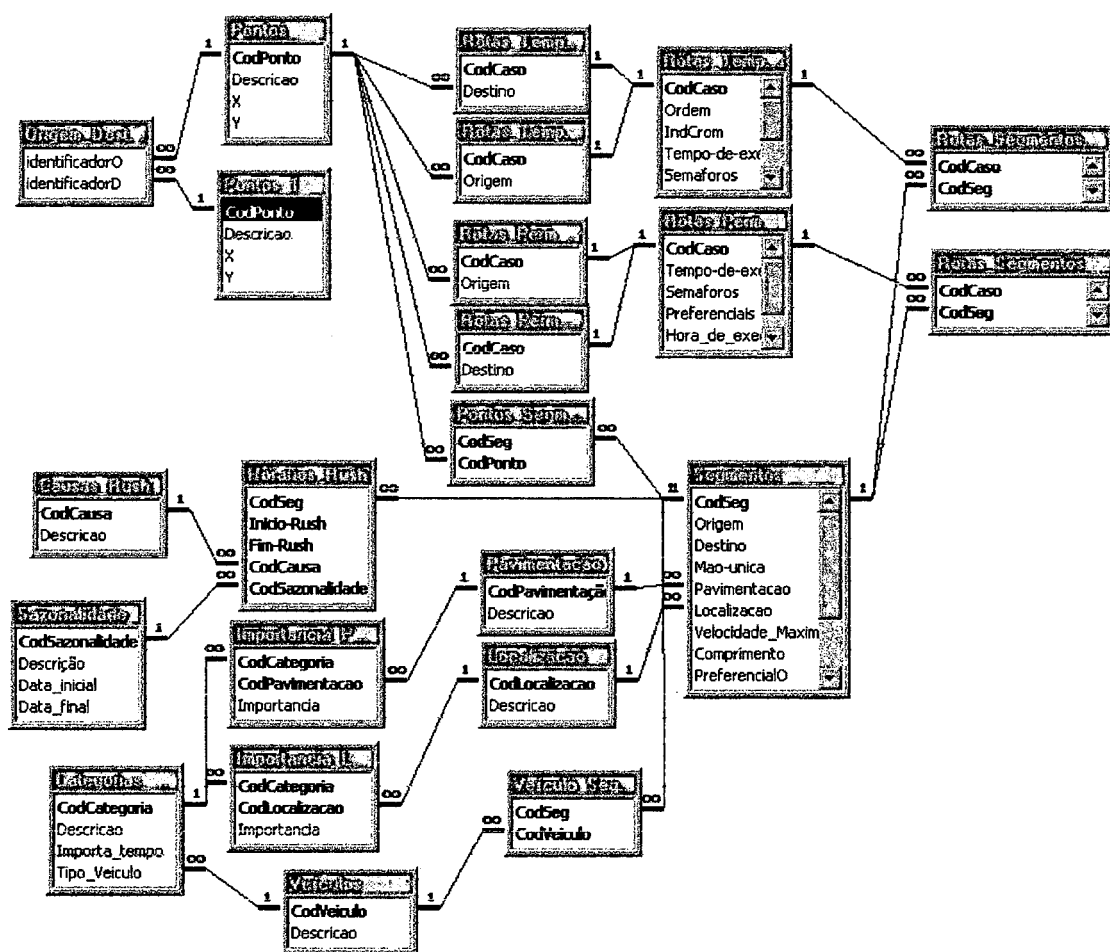
WHITLEY, D.; STARKWEATHER, T.; SHANER, D.. The Travelling Salesman and the Sequence Scheduling: Quality Solutions Using Genetic Edge Recombination. In: **Handbook of Genetic Algorithms**. 1^a. ed. New York: 1991. p. 350-372.

WILLUMSEN, L.; HOUNSELL, N. B., Simple Models of Highway Reliability – Supply Effects. . In: SEVENTH INTERNATIONAL CONFERENCE ON TRAVEL BEHAVIOUR (IATBR-1994). Santiago: jun. 1994. p. 65-76.

WOOLDRIGE, M.; JENNINGS, R. N.. Intelligent Agents: Theory. In *Knowledge engineering Review*, October, 1994. Revised in January, 1995.

XU, L., D.. Na integrated rule- and case-based approach to AIDS initial assessment. In **International Journal of Bio-Medical Computing**, 40. p. 197-207, 1996.

ANEXO 1 - DIAGRAMA ENTIDADE RELACIONAMENTO



ANEXO 2 - ALGORITMO

Corpo Principal

Início

```

Inicialização ( )
Sucesso = True
Recuperação de Rotas Completas ( )
Se Recuperação de Rotas Completas = Executada com Sucesso então
    Cálculo da Similiaridade das Rotas ( )
    Se Valor de Similaridade = Aceitável então
        Estabelecimento do Ranking para três melhores ( )
        Plotagem das Rotas ( )
    Senão
        Opção de Adaptação ( )
        Se Adaptação = Executada com Sucesso então
            Montagem da rota com o trecho adaptado ( )
            Plotagem das Rotas ( )
Senão
    Recuperação de Trechos de Rotas
    Se Recuperação de Trechos de Rotas = Executada com Sucesso então
        Cálculo da Similaridade das Rotas ( )
        Se Valor de Similaridade = Aceitável então
            Estabelecimento do Ranking para três melhores ( )
            Plotagem das Rotas ( )
        Senão
            Opção de Adaptação ( )
            Se Adaptação = Executada com Sucesso então
                Montagem da rota com o trecho adaptado ( )
                Plotagem das Rotas ( )
    Senão
        Módulo Genético ( )
        Se Geração de Rotas = Sucesso então
            Cálculo da Similaridade das Rotas ( )
            Estabelecimento do Ranking para três melhores ( )
            Plotagem das Rotas ( )
        Senão
            Sucesso = False
Se Sucesso então
    Opção de Aprendizado ( )

```

Fim.

Procedimentos e Funções

Inicialização ()

Início

```

Limpeza a base de casos temporária ( )
Inicialização das variáveis para o Módulo Genético ( )
Inicialização das variáveis para o Módulo RBC ( )

```

Fim

Recuperação de Rotas Completas ()

Início

Busca na Tabela de Origem e Destino das Rotas, procurando uma rota que possua a Origem e o Destino, especificados pelo usuário ()

Fim

Cálculo da Similaridade das Rotas ()

Início

Para cada Rota Recuperada faça

Somatória/normalização dos pesos de suas características para cada segmento, de acordo com o perfil do usuário requisitante;

Cálculo do Valor de Similaridade a partir da Métrica de Similaridade definida

Fim

Fim

Estabelecimento do *Ranking* para três melhores ()

Início

Selecionar as três rotas que possuem o três maiores e distintos Valores de Similaridade

Fim

Plotagem das Rotas ()

Início

Para cada Rota faça

Para cada Segmentos da rota faça

Pinte o trecho do mapa correspondente ao segmento

Fim

Fim

Fim

Opção de Adaptação ()

Início

Se Primeira Vez = True então

Primeira Vez = False

Reinicia-se o processo do RBC – procurando por uma rota que satisfaça o trecho que se deseja adaptar ()

Senão

Adaptação não Permitida

Fim

Recuperação de Trechos de Rotas ()

Início

Busca na Tabela de Rotas_Segmentos procurando uma rota que possua segmentos adjacentes à Origem e ao Destino, especificados pelo usuário ()

Fim

Opção de Aprendizado ()

Início

Se (Rota = Gerada/Adaptada) e (Desejo do usuário = Armazenar) então
Gravar a Rota executada na Base de Casos Permanente

Fim

Módulo Genético ()

Início

MaxPop = número máximo de indivíduos projetados (constante)

MaxGen = número máximo de gerações projetadas (constante)

Gen = 0 (contador de gerações)

RotasCompletas = *logic* (a rota alcançou o destino?)

Para I := 1 **até** MaxPop **faça**

Repita Mutation

 Candidata (tentar mudar '1's para '*')

 Reset(mudanças randômicas no alelos '1' e '0')

Até (RotasCompletas) **ou** (++Gen = MaxGen)

Fim

Fitness (ordenação da população de acordo com o comprimento)

Gravar Rotas Completas na Base de Casos Temporária

Fim

***ANEXO 3 – ARTIGO PUBLICADO REFERENTE AO
TRABALHO.***

A Hybrid System For Route Planning

Sarajane Marques Peres, Josué Pereira de Castro

Alejandro Martins, Roberto C. S. Pacheco

Aran Bay Tcholakian, Rosina Weber Lee

Lia Caetano Bastos, Fernando Gauthier

Graduate Program in Production Engineering - PPGE

Federal University of Santa Catarina, Brazil, UFSC

martins@eps.ufsc.br

Abstract

This paper presents a hybrid system for route planning. The system is composed of two principal modules: a case based reasoning (CBR) and a genetic module. The CBR module is responsible for the user interface. It retrieves the solutions founded by the genetic module, and adapts these solutions to the user preferences. The genetic module is responsible for generating new routes, based on the shortest path between the origin and destination points.

Introduction

Several research areas have addressed the Route Planning optimization problem, including Geographic Information Systems (Abdel-Aty, Abdallah and As-Saidi, 1997), Decision Theory, Genetic Algorithms (GA), Operational Research (Golden and Assad 1988), and Case-based Reasoning (CBR). We propose the combination of GA and CBR to improve route planning efficiency. An effective solution toward this end should deal with constraints such as rush hour delays, traffic patterns, road conditions, and so on. By integrating different artificial intelligence models we have designed a system that targets such objective.

The system is composed of two modules: the case-based reasoning and the genetic module. The first is responsible for the user interface. It evaluates the routes stored in a case base and executes the best retrieved solution. If none adequate route is retrieved, the genetic algorithm module is activated to create new routes. When the GA does not meet any feasible route, the system proceeds by adapting one of the similar solutions found in the case base.

Route Planning

Route planning is an optimization task with a variety of applications. Several authors have dealt with this subject. Lee and Fishwick (1995), for instance, developed simulation techniques to automate the decision making process in

uncertain and complex environments. Christofides (1985) also describes an algorithm and several heuristics to solve vehicle routing problems.

The proposed system aims to help drivers to decide the route to be taken between two given points in urban zones. The route chosen must be both efficient (regarding minimal feasible length) and satisfactory (in terms of driver's preferences). Meeting both objectives has been neglected in route planning literature, making optimization techniques insufficient to balance optimality and user's satisfaction. The combination of CBR and GA made possible to find routes where both aspects are considered.

CBR and GA in Related Applications

Case-based reasoning and genetic algorithms have already been applied in traffic control and route planning, either as combined or stand-alone approaches.

Nakamiti and Gomide's work (1994) have applied CBR and GA to manage traffic flow more efficiently. Given a traffic condition, the system controls the light times by adapting successful solutions given for previous similar conditions. The system uses GA to adapt the retrieved cases from the case base. The retrieval is based on attribute similarity between the input and the stored cases.

Haigh and Veloso (1995) developed a case-based reasoning system for route planning method. The routes used in the past are stored and may be retrieved and reused to generate new routes. In this system, the planner can use several retrieved cases to generate a new route by merging these cases. An *efficiency value* is associated to each retrieved case, indicating its "quality". Depending on these values, the system either reuses the known routes or searches for alternative routes.

Choi and Woo (1997) proposed an evolutionary route planning algorithm to reveal the optimal route between origin and destination

in road traffic networks. Their work also presents a simulation of a network with several

constraints.

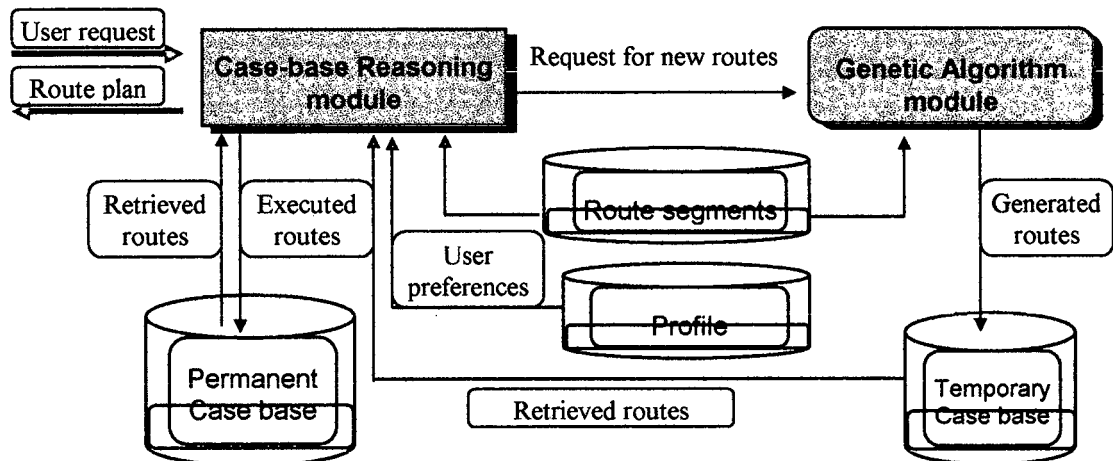


Figure 1: System Architecture.

The System Modeling

Balancing optimality and user satisfaction requires a model that keeps tracking of both targets while seeking for a solution. This can be understood as the need for *general knowledge*, that is, the model has to balance requirements coming from different frameworks. In CBR theory, a system that employs an external method of general knowledge is classified as a horizontal integrating system (Aamodt, 1993). In order to provide general knowledge we combine CBR with GA (improving the search for intended results). The system is then classified as a *hybrid system*, since it combines two AI techniques on a single horizontal integrating system (Medsker, 1995).

The route site (street traffic network) is represented by road segments and their intersections. Each route is composed by a subset of adjacent and ordered segment. The route begins by the initial point of the first segment (origin) and ends with the final point of the last segment (destination).

The system uses the following parameters: (a) *the user's profile*, consisting of an archive with restrictions (e.g., site restrictions, street conditions preferred, etc.) for each user category (i.e., ambulance service, taxi drivers, etc.); *the user's request* (origin and destination and traffic time). The user's request is converted into a case, firing the CBR module. This module searches for similar cases (eventually adapting the most similar routes to the input case), presenting the solutions to the user. If no similar case is found, the genetic module is activated.

When requested, the genetic module creates

new routes converted into cases and returns them to the CBR module. The CBR, then, proceeds the adaptation to the input case. Once again, the user checks the solution. If, after this process, a satisfactory route was not found, the system displays the restrictions prohibiting the route generation. The system architecture is briefly described in Figure 1.

The Case-based Reasoning Module

The first requirement in the proposed model is to meet the user profile. Humans tend to keep or adapt previous successful approaches when facing new similar situations. This was the main motivation for building a Case-based Reasoning module to pursue a satisfactory route plan, applying analogical reasoning. This module is responsible for presenting route suggestions to the user based on his or her request. Figure 2 depicts the route representation used in the CBR module. Some attributes are used as indexes and others as route descriptors. The case indexes are the following: origin and destination points, segments, case suitability degree, and user's profile. The other attributes are only descriptive and are presented to the user as additional information.

ID	Origin	Dest	Adequate Degree	Descriptive Attributes	ID	Segments

Figure 2: Route representation in the case base.

Initially, the module searches the case base for the most similar route to the one desired by the user. Eventually, there is need for performing a route adaptation (e.g., traffic accident). In this case, the adaptable route has segments that can be

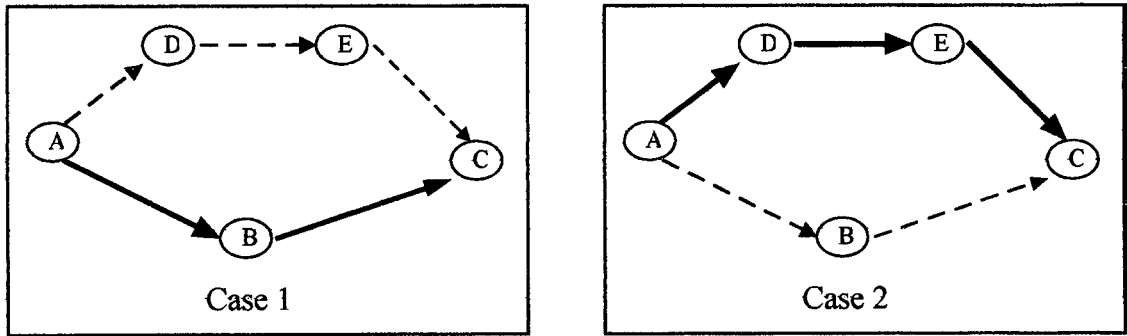


Figure 3: Route adapting process.

replaced by another set of *satisfactory* segments. The substitutive set of segments is obtained from a partial case retrieval. Figure 3 illustrates an example of adaptation process. The interval ABC can be replaced by the interval ADEC, observed in a different case.

If no satisfactory solution is found, the genetic module is activated, a new temporary case base is created, restarting the retrieving and adapting processes.

If the route selection was successful, the system runs the solution. Afterwards the user chooses whether or not include the route in the case base.

One of the most significant aspects in route planning is the need for rapid adaptation due to traffic network updating. In this system, the CBR module makes adaptive learning possible. In the beginning, the case base is empty. With its utilization, the routes are created, run, and stored in the case base, characterizing a continuous learning process.

The Genetic Module

The GA role is to find feasible routes between the user's origin and destination points. The strategy consists in searching and evaluating alternatives, seeking for the shortest route path.

The first modeling issue is how to represent routes as binary codes (*i.e.*, GA chromosomes). Previous work (*e.g.*, Whitley, Starkweather and Shaner, 1991) have addressed this matter by allowing every two-point combinations. This is valid only for networks fully connected. Actual routes require that every segment is a feasible path. This makes the usual mutation and cross-over operations inadequate, since they can lead to non-existent segments. In this work, we have developed an order independent scheme that makes the GA operations applicable in non-fully connected networks. This scheme allows the modeling of oriented graph and also guarantees route feasibility.

The GA module begins by establishing a random set of routes. Each route is a set of

oriented segments, created by gradually adding paths. The addition of segments is guided by the shortest Euclidean distance between the candidate nodes and the destination point. The route assembling continues until the algorithm reaches either a final segment (hitting the destination) or the maximum number of generations. GA proceeds attempting to establish more routes, until reaching a fixed number of runs (population size).

In this module, each route is represented as a chromosome (see Figure 4), where:

- a street traffic network segment is modeled by an allele;
- each allele receives one of the values in {0,1,*,#}. Value '0' means that the segment, at the current iteration, does not belong to the route; '1' indicates that the segment is a path candidate; '*' indicates a route segment; and '#' marks the segment as unfeasible for the current route;

1	2	3	4	5	6	7	8	9	10	...	N
0	*	0	#	*	1	*	#	1	0		0

Figure 4: Route represented in a chromosome form.

The *fitness operator* is applied to every allele valued with '1'. It is defined by three operations: (a) calculus of the Euclidean distances between all candidates and the destination; (b) *feasibility test*, which verifies each segment feasibility; and (c) *evaluation*, which checks for either change the allele value (to '*' or '#') or keep it as a candidate, according to the following criteria:

- If the candidate cannot be connected to the current segment set (*i.e.*, it is an unfeasible segment), take its Euclidean distance to the destination (d_c). Take also the shortest Euclidean distance between the current node set and the destination (s_d) (*i.e.*, the shortest Euclidean distance among the '*' alleles). If $d_c > s_d$, mark the candidate allele with '#'. This means that this segment will definitely not belong to the current route.
- Among all feasible candidates, take the one

with minimum Euclidean distance to the destination and convert its allele value to '*'.

The *mutation* operator works on each allele valued with either '0' or '1', randomly deciding whether to change it or not to its opposite value. The alleles valued with '*' or '#' remain unchangeable, fixing the schemata in order to assure segment feasibility and the algorithm efficiency in further operations.

The algorithm in Table 1 describes the genetic module operations:

Table 1: Algorithm used in the GA module.

```

MaxPop = max number of desired individuals
(constant)
MaxGen = max number of desired generations
(constant)
Gen = 0 {generation counter}
CompleteRoute = boolean {does the route hits the
destination?}
For I := 1 to MaxPop do
  Repeat
    Fitness {try to change all '1' to '*' or '#'
alleles}
    Mutation {only in '0' and '1' alleles}
    Until (CompleteRoute) or (++Gen = MaxGen)
  End

```

The GA output is an ordered set of optimized routes. The single criteria for discovering these routes was the shortest path. The other user's criteria are still missing. In order to balance length and satisfactory aspects, the GA responses have to be migrated to the CBR module. This is performed by transforming the GA outputs into a case base format (temporary case base). Depending upon the user decision, the temporary cases are incorporated into the permanent case base by the CBR module:

Case Study

As an example of the system application, we choose a ambulance service, whose principal restriction is the execution speed of the route. In time critical problems, it is important to avoid paths where traffic is influenced by daytime. For this reason, we chose the route execution hour and day as the attributes in the profile archive. Figure 5 depicts an actual street traffic network for the ambulance.

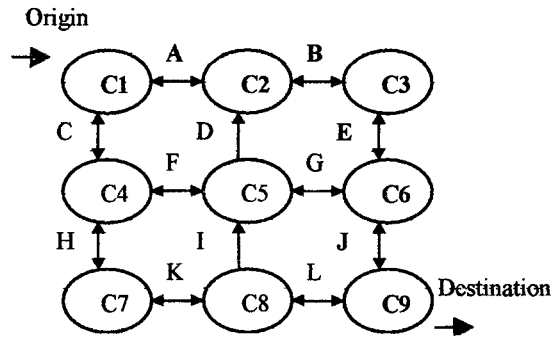


Figure 5: Example of street traffic network.

First, the user (ambulance driver) asks for a route plan. The request is evaluated by the CBR module, which seeks for similar cases. In this case, the suggested route was 'A-D-I-L'. The user was unsatisfied by this route plan. He knows that segment 'L' is currently blocked. The CBR module calls the GA module, asking for alternative solutions, and marking the segment L as unfeasible in the street traffic network profile.

A	B	C	D	E	F	G	H	I	J	K	L
*	1	1	0	0	0	1	0	1	0	0	#

Figure 6: First individual of the population.

As shown in Figure 6, the GA module begins by establishing an first individual of the population (i.e., a set with the first route segment and other candidates).

The next step is to evaluate this individual, searching for the segment feasibility (Figure 7). In this case, segment B is the only feasible path. The algorithm also analyzes whether the unfeasible segments can eventually belong to the current route. In this case, segment C will never belong to any route containing segment A. Thus, this segment is made permanently unfeasible (i.e., it is marked with '#').

A	B	C	D	E	F	G	H	I	J	K	L
*	*	#	0	1	1	0	0	0	1	0	#

Figure 7: Individual after fitness followed by mutatic

The route design process remains until a final segment is found or until the algorithm hits the maximum number of generations. Figure 8 shows the result after a sequence of GA iterations over the same route plan.

A	B	C	D	E	F	G	H	I	J	K	L
*	*	#	0	*	0	1	1	1	*	0	#
1	2			3					4		

Figure 8: Complete and ordered route obtained after some fitness and mutation operations.

The next GA step is to convert the suggested route plan into a case and send it to the temporary case base. The result is shown in Figure 9.

ID	Source	Destination	Adeq. Degree	Hour	Day	weather Conditions
1	C1	C9	90%	12:00	Mon	Good

(a)

ID	Segment
1	A
1	B
1	E
1	J

(b)

Figure 9: (a) Temporary case base; (b) Route adjusted in a form of a case.

Finally, the CBR module takes place and presents the new route to the user. If he decides to take the suggestion, the CBR integrates the new route into the permanent case base for future use in similar situations.

Conclusions and Future Work

In this work, we have presented a new model for route planning problems. In this approach, the suggested route plans are both efficient and satisfactory. They represent a balance between optimality (minimal feasible length) and user requirements. This was made possible by means of a hybrid intelligent system, including a module for each purpose. The user's requirements are pursued by the Case-base Reasoning module, while optimality is seek by the Genetic Algorithm module.

Analogical Reasoning and adaptive learning were the main motivation for building a CBR. This module not only takes similar past experiences to propose route plans, but also uses the new information to increase its memory for further iterations. When a solution is not available (insufficiency of knowledge or user dissatisfaction), CBR calls the Genetic Algorithm module.

The GA purpose is to establish route plans meeting the new conditions and keeping optimality as the primer concern. In addition, the method had to keep feasibility constraints and boost optimality search. We achieved these objectives developing a new chromosome representation scheme, suitable for route plan problems in oriented graph networks. The found route plans are evaluated by the CBR and presented to the user. The new route plan is eventually incorporated in the permanent case base.

The interaction between the CBR and GA aims to balance satisfaction and optimality constraints. By keeping the objectives separate and integrated we avoid increasing complexity in each module

(e.g., by including user profile in the GA, one would increase iteration time, since the fitness would be a more complex operator). We have also made adaptive learning possible, since the case base is dynamic with the time.

As future work, the group is evaluating different alternatives. First, we intend to implement an interface with a Geographic Information System (GIS). This optimizes the addition of new street-traffic network configurations. We are also studying the possibility of including a fuzzy expert system to deal with season variables (e.g., tourist season, concerning school, etc.) and temporal data (e.g., daytime), related to traffic flow conditions. An example of combining GIS and fuzzy modeling can be found in Lee and Lee's work (1996). Finally, we are also studying other approaches for the optimality search, such as the use of GA* (Logan and Poli, 1997).

References

- Abdel-Aty, M.A., Abdallah, M.N. and As-Saidi, A. H. 1997. A Methodology for Route Selection and Guides Using GIS and Computer Network Models. In *Proceedings of 76th Annual Meeting of The Transportation Research Board*, 20-21. Washington D.C.: Transportation Research Board.
- Aamodt, Agnar. Explanation-Driven Case-Based Reasoning. *Topics in Case-Based Reasoning, (First European Workshop, EWCBB-93)*. Wess, Stefan, Althoff, Klaus-Dieter & Richter, Michael (editors) Springer-Verlag, 274-288, 1993.
- Choi, G.S. and Woo, K.D. 1997. The development of Optimal Route Algorithm Based On Evolution Program. *The Transactions of The Korean Institute of Electrical Engineers* 46(2):294-297
- Christofides, N. 1985. Vehicle Routing In The Traveling Salesman Problem - In *A Guided Tour of Combinatorial Optimization*, 431-448. Great Britain.: John Wiley & Sons Ltd.
- Golden, B. L., and Assad, A., 1988. *Vehicle Routing: Methods and Studies*. Amsterdam, Netherlands: Elsevier Science Publishers B.V.
- Haig, K. Z. & Veloso, M., 1995 Route planning by analogy. *Proceedings of Case Based Reasoning research and development, First International Conference: ICCBR-95*, Springer-Verlag, 171-180.
- Lee, J. and Fishwick, P.A., 1995. Simulation Based Real Time Decision Making for Rout Planning. *1995 Winter Simulation Conference*, 1087-1095. Crystal City, VA.
- Lee, H.C., and Lee, C.C. 1996. Inexact strategy and Planning in Taipei City. In *Proceedings of*

Asian Fuzzy Systems Symposium, 1996, Kenting, Taiwan, Soft Computing in Intelligent Systems and Information processing, 308-313.
Piscataway, NJ.: IEEE Service Center.

Logan, B and Poli, R. 1997. Route Planning with GA*, Technical Report, CSRP-97-17, School of Computer Science, University of Birmingham, England

Medsker, L. R. , *Hybrid Intelligent Systems*, Kluwer Academic Publishers: Boston, 1995.

Nakamiti, G. & Gomide, F., An evolutive fuzzy mechanism based on past experiences,. *Second European Congress on Intelligent Techniques in Soft Computing*, Aachen, Germany, sept, 1994.

Whitley, D., Starkweather, T. and Shaner, D. 1991. The Traveling Salesman and the Sequence Scheduling: Quality Solutions Using Genetic Edge Recombination. In *Handbook of Genetic Algorithms*, 350-372. New York, NY.: Van Nostrand Reinhold.